

**SMARTLMS - A ROLE BASED AI-ASSISTED LEARNING SYSTEM FOR
STUDENTS AND LECTURERS**

BY

MACAULAY OMOIKHOJE DE-GREAT

PSC2105355

**DEPARTMENT OF COMPUTER SCIENCE,
FACULTY OF PHYSICAL SCIENCES,
UNIVERSITY OF BENIN,
BENIN CITY, EDO STATE,
NIGERIA.**

NOVEMBER, 2025.

**SMARTLMS - A ROLE BASED AI-ASSISTED LEARNING SYSTEM FOR
STUDENTS AND LECTURERS**

BY

MACAULAY OMOIKHOJE DE-GREAT

PSC2105355

**A PROJECT REPORT SUBMITTED TO THE DEPARTMENT OF
COMPUTER SCIENCE, FACULTY OF PHYSICAL SCIENCE,
UNIVERSITY OF BENIN, BENIN CITY, EDO STATE, NIGERIA.**

**IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE
AWARD OF A BACHELOR OF SCIENCE (B.Sc.) DEGREE IN COMPUTER
SCIENCE.**

NOVEMBER, 2025.

CERTIFICATION

This is to certify that this project work was carried out and done by **MACAULAY OMOIKHOJE DE-GREAT** with Matriculation Number **PSC2105355** under my supervision. It is adequate and satisfactory, both in scope and content, for the award of Bachelor of Science (B.Sc.) Degree in the department of Computer Science, Faculty of Physical Science, University of Benin.

MR. JOSEPH OKHUOYA

Project Supervisor

Date

APPROVAL

This project work is hereby approved in partial fulfilment of the requirements for the award of Bachelor of Science (B.Sc.) Degree in Computer Science from the University of Benin.

MR. JOSEPH OKHUOYA

Project Supervisor

Date

Dr. (Mrs.) A. R. Usiobaifo

Head of Department (HOD)

Date

DEDICATION

I humbly dedicate this project to God Almighty for His inspiration and guidance, strength and wisdom to see through to the completion of this project. I also dedicate this project to my dad and family, Engr Aitokhaimen Ejekhile for his unending and selfless support and guidance.

ACKNOWLEDGEMENT

First and foremost, I am deeply grateful to God Almighty, whose grace, strength, and wisdom have brought me this far. Without His divine support, this academic milestone would not have been possible.

I would like to express my sincere appreciation to my project supervisor, Mr. Joseph Okhuoya for his insightful feedback, explanations, and support which really helped me a lot in writing my project report. Special thanks also to our project coordinator, Dr Maxwell Osagie for his activeness and prompt support to all of us during the course of this project.

My heartfelt gratitude goes to my dad, Engr. Aitokhaimen Ejekhile for his unending support and steadfast encouragement. Your belief in me has always been a source of strength and motivation.

I also want to appreciate my friends, course mates, and everyone who contributed in one way or another. Your support, encouragement, and positive energy really meant a lot.

ABSTRACT

This project focused on the design and implementation of SmartLMS, an Artificial Intelligence assisted Learning Management System (LMS) developed to enhance digital teaching and learning in the University of Benin. The aim of the study was to create a flexible and user-friendly platform that integrates AI features to support both lecturers and students in managing learning materials, assessments and performance analytics. The system was designed using Next.js for the Frontend, Node.js with the Express.js framework for the Backend and MongoDB as the database, with JWT authentication to regulate signup and secured login to the platform, Cloudinary to handle and store uploaded materials and OpenAI and Gemini AI for AI-driven functionalities. The AI modules were implemented to help students with course materials summarization, self-practice test generation and keep track of students' analytics. The system followed a modular design and was implemented to be responsive, reliable and accessible on both mobile and desktop devices. Results showed that SmartLMS improved teaching efficiency, student engagement and learning personalization while maintaining lecturer control over assessments, grading and students' academic performance. The project concluded that SmartLMS provides a practical and scalable solution for integrating AI into education, offering a more interactive, intelligent and accessible learning experience that can be integrated into different learning environments.

TABLE OF CONTENTS

CERTIFICATION	i
APPROVAL	ii
DEDICATION	iii
ACKNOWLEDGEMENT	iv
TABLE OF CONTENTS	vi
LIST OF FIGURES	ix
CHAPTER 1	1
INTRODUCTION	1
1.1 BACKGROUND OF THE STUDY	1
1.2 STATEMENT OF THE PROBLEM	3
1.3 AIMS AND OBJECTIVES	4
1.4 SIGNIFICANCE OF THE STUDY	5
1.5 SCOPE OF THE STUDY	7
1.6 LIMITATIONS OF THE STUDY	8
1.7 DEFINITION OF TERMS	9
CHAPTER 2	11
Literature Review	11
2.1 BACKGROUND	11
2.2 CONCEPTUAL REVIEW	12
2.2.1 LEARNING MANAGEMENT SYSTEMS	12
2.2.2 ARTIFICIAL INTELLIGENCE IN EDUCATION (AIED)	13
2.2.3 SMART LEARNING SYSTEMS	13
2.2.4 MOBILE AND CROSS-PLATFORM LEARNING SYSTEMS	14
2.3 THEORETICAL REVIEW	14
2.3.2 COGNITIVE LOAD THEORY	15
2.3.3 SELF-REGULATED LEARNING THEORY	15
2.3.4 CONNECTIVISM THEORY	16
2.4 EMPIRICAL REVIEW	17

2.4.1 INTERNATIONAL STUDIES	17
2.4.2 NATIONAL STUDIES (NIGERIA)	18
2.4.3 LOCAL STUDIES (UNIVERSITY CONTEXT)	19
SUMMARY OF FINDINGS	19
2.5 REVIEW OF EXISTING LEARNING MANAGEMENT SYSTEMS	20
2.5.1 MOODLE	20
2.5.2 BLACKBOARD	21
2.5.3 CANVAS	21
2.5.4 EMERGING AI-ENHANCED LMS TRENDS	22
2.6 RESEARCH GAP	22
2.7 SUMMARY OF LITERATURE REVIEW	24
CHAPTER 3	27
METHODOLOGY AND SYSTEM ANALYSIS	27
3.1 BACKGROUND	27
3.2 RESEARCH DESIGN	27
3.3 SYSTEM DEVELOPMENT METHODOLOGY	28
3.4 SYSTEM ANALYSIS	30
3.4.1 EXISTING SYSTEM ANALYSIS	30
3.4.2 PROBLEMS OF THE EXISTING SYSTEM	30
3.4.3 PROPOSED SYSTEM ANALYSIS	31
3.4.4 SYSTEM REQUIREMENTS	32
3.5 SYSTEM DESIGN	33
3.5.1 SYSTEM ARCHITECTURE	33
3.5.2 USE CASE DIAGRAM	35
3.5.3 ENTITY RELATIONSHIP DIAGRAM (ERD)	37
3.5.3 USER INTERFACE DESIGN	38
CHAPTER 4	40
SYSTEM IMPLEMENTATION	40
4.1 BACKGROUND	40

4.2 SYSTEM IMPLEMENTATION	40
4.2.1 FRONTEND IMPLEMENTATION	41
4.2.2 BACKEND IMPLEMENTATION	42
4.2.3 DATABASE IMPLEMENTATION	42
4.2.4 AI INTEGRATION	43
4.2.5 HOSTING AND DEPLOYMENT	43
4.3 SCREENSHOTS OF THE SOFTWARE	44
CHAPTER 5	63
SUMMARY, CONCLUSION AND RECOMMENDATION	63
5.1 BACKGROUND	63
5.2 SUMMARY OF THE FINDINGS	63
5.3 CONCLUSION	64
5.4 RECOMMENDATIONS	64
5.5 SUMMARY	65
REFERENCES	66

LIST OF FIGURES

Figure 3.1 SmartLMS System Architecture (Three-Tier Design)	34
Figure 3.2 SmartLMS Use Case Diagram	36
Fig 3.3 SmartLMS Entity Relationship Diagram (ERD Diagram)	38
Fig 4.1 Registration page	44
Figure 4.2 Login page	44
Figure 4.3 Forgot Password Page	45
Figure 4.4 OTP Verification Page	45
Figure 4.5 Set New Password Page	45
Figure 4.6 Lecturer Homepage Page	46
Figure 4.7 My Courses Page Shows Courses Created by Lecturer	46
Figure 4.8 Lecturer creating a new course	47
Figure 4.9 The course details page showing information about each course. The materials, assessments and announcements	48
Figure 4.10 Uploading materials (different file formats) for students	49
Figure 4.11 Create an assessment (Quiz, tests, CA) for students	50
Figure 4.12 Grading Page for published assessments	51
Figure 4.13 Analytics page where lecturer tracks the performance of students for each assessment and can export the analytics report in different format	52

Figure 4.14 Page for Lecturer create announcements for students	53
Figure 4.15 Messaging and Feedback page for easy communication between lecturers and students	54
Figure 4.16 Profile page for lecturer to edit their personal information	55
Figure 4.17 Student Homepage	56
Figure 4.18 Student Enrolled Courses	57
Figure 4.19 Profile page for lecturer to edit their personal information	58
Figure 4.20 Student Assessment Page	59
Figure 4.21 Uploaded Materials Page	60
Figure 4.22 Student Sample in the Database	61
Figure 4.23 Course Sample in the Database	61
Figure 4.24 Course Material Sample in the Database	62
Figure 4.25 Enrolled Students	62

CHAPTER 1

INTRODUCTION

1.1 BACKGROUND OF THE STUDY

Digital tools are now part of everyday student life. With many learners accessing course content from phones, tablets and laptops, universities have adopted Learning Management Systems (LMSs) and a growing set of AI tools to support learning from automated summaries and practice questions to dashboards that flag struggling students (Wang et al., 2024). But having a collection of features does not guarantee practical usefulness. In many institutions, students and teachers still juggle email threads, WhatsApp groups, cloud folders and a central LMS that can feel clunky or fragmented. At the same time, lecturers spend substantial time preparing materials and interpreting student data. This mismatch between the potential of AI and how well systems actually fit everyday teaching workflows creates an opportunity for focused improvement.

Recent reviews of AI in education highlight significant promise: chatbots, intelligent tutoring, and analytics can increase engagement, personalize learning and reduce routine work (Abdallah et al., 2024; Wang et al., 2024). Yet the literature also identifies familiar problems: siloed tools, inconsistent user experiences across devices, limited teacher control over AI outputs, and weak alignment with local curricula. These shortcomings make many “AI features” feel experimental rather than genuinely helpful in the classroom.

Local data from the University of Benin illustrates this gap. A recent survey there found broad awareness of AI among students and general openness to AI-assisted learning, but also pointed to limited practical adoption because campus systems are fragmented and not always convenient

for everyday use across devices (Chizobam, 2024). Lecturers reported wanting aids that save preparation time and give clearer, actionable insight into student learning provided those aids retain academic oversight and do not replace human judgment.

This study responds with a practical, teacher-centred SmartLMS pilot built to work smoothly on both desktop and mobile platforms. The system offers AI-assisted features automated lecture summaries, suggested note points, AI-generated practice questions and candidate test items, and student engagement and performance analytics to track student progress and assessments records (these assessments includes tests, CAs and assignments) but it is explicitly assistive rather than autonomous. Instructors will receive AI-generated drafts and analytics as editable suggestions; final assessment, feedback and grading remain the instructor's responsibility. This design choice preserves academic control while reducing routine workload and improving access to study resources.

Starting the pilot in the Computer Science department is strategic. Computer Science students and staff tend to be early adopters who comfortably use digital tools; the department's structured curriculum also makes it easier to test and refine AI prompt templates and analytics metrics. Running a focused pilot enables quick iteration with lecturers and students, clearer evaluation of learning outcomes, and a practical path to scale the system to other departments once workflows, privacy safeguards and instructor controls are validated.

More than another feature list, this project aims to fix the weaknesses highlighted in the literature and the UNIBEN context: deliver a cohesive, responsive student experience across devices; provide teacher-centred AI outputs that save prep time without undermining human judgment; and align outputs with local syllabi and assessment styles. If successful, the SmartLMS pilot will

show a practical model for integrating AI into campus learning one that improves student access to practice and revision tools, gives lecturers timely analytics and suggested resources, and maintains the lecturer's final authority on grading and assessment.

1.2 STATEMENT OF THE PROBLEM

Students at the University of Benin are already living in an AI world. Many say they're comfortable with AI tools and often turn to informal channels like WhatsApp and Facebook to get help, share study notes, or find practice questions (Chizobam, 2024). That sounds promising but it also means students are piecing together support from scattered places rather than using one dependable system that's built around their courses.

Researchers have shown that AI can do amazing things for learning: personalized practice, instant summaries, chat-based help, and analytics that spot who's falling behind (Wang et al., 2024; Abdallah et al., 2024). Yet most of the exciting studies stop at prototypes or lab experiments. What's missing is a practical, real-world system that brings those AI features into everyday teaching in a way instructors trust and control. In other words, we have tools that can help, but not a trusted hub that fits into how lecturers actually teach and grade.

That gap creates concrete problems for a department like Computer Science. Students don't have a single place to get AI-generated practice tests, smart summaries, or clear progress reports. Lecturers don't have an easy, transparent way to generate practice materials or see analytics without handing over final grading decisions to an algorithm. And administrators and staff worry about privacy, bias, and whether these AI features will actually work in real classes (Wang et al., 2024).

So, the problem is simple but important: we need an applied, ethically-aware SmartLMS that pulls useful AI tools together practice/test generation, AI summaries and notes, assessment support, and performance analytics while keeping lecturers firmly in charge of formal grading. Build that for one department first (Computer Science), and it could scale to the whole school. That's the practical improvement this project aims to deliver: useful AI, joined up and human-centered, not AI for its own sake.

1.3 AIMS AND OBJECTIVES

The aim of this study is to design, build and evaluate **SmartLMS**, a responsive learning platform (mobile + desktop) for students in the University of Benin that adds AI-powered tools to support teaching and learning. This AI-powered tools make it possible and easy to generate practice and test questions from uploaded materials from lecturers, generate summary notes and short notes from bulky materials. smartLMS also have basic features such as the ability for lecturers to upload any type of course materials (pdfs, docs, videos, links, images, etc) to the classroom at any time. smartLMS will also provide lecturers with the feature to set assessments (assignments, tests and CA) on the platforms and grade them. Lecturers are all in control of al assessments and grading systems with access to performance analytics to track the academic progress of their students.

The objectives are as followed:

- i. To design and develop a SmartLMS platform with a simple, scalable architecture and responsive interface that works effectively across both mobile and desktop devices, integrating AI modules for generating practice questions, summaries, and short notes to enhance a personalized learning experience for students.

- ii. To implement lecturer and student functionalities that support course management, manual grading, assessment assistance, performance analytics, and privacy safeguards to ensure responsible use of AI in academic activities.

1.4 SIGNIFICANCE OF THE STUDY

This study is very important and its significance touches many categories and they are enumerated below:

1. For students

- ❖ SmartLMS will make studying less stressful and more organized. With features like quick summaries, practice tests to test their knowledge, and clear note points, students won't always need to dig through long slides or freak out about having to finish reading bulky materials before exams (Zawacki-Richter et al., 2019).
- ❖ It gives learners room to move at their own pace, which is especially helpful in big classes where lecturers cannot give one-on-one guidance to everyone (Costa et al., 2017).
- ❖ Students also receive personalized feedback from lecturers after each returned assessments based on their performance.
- ❖ Students can take practice tests and assessments and view their performance and see for themselves how well they are prepared for exams.
- ❖ Regular feedback and progress tracking will also keep students motivated and more involved in their learning.

2. For lecturers

- ❖ The platform helps cut down repetitive tasks such as preparing practice materials or condensing lecture content, allowing lecturers to focus more on teaching and mentoring (Celik et al., 2022).
- ❖ With the analytics dashboard, lecturers can easily track students' performance and see which students are struggling or excelling, making it easier to intervene where necessary (Charitopoulos et al., 2020).
- ❖ With this platform, lecturers are able to give announcements, which can include impromptu tests update, presentations and class updates and every student will get it.
- ❖ Most importantly, lecturers still remain in charge of grading and academic decisions. The system only assists, ensuring that trust and authority stay with them (Akgun & Greenhow, 2022).

3. **For the University of Benin (After further improvements)**

- ❖ SmartLMS provides a locally designed solution that actually fits Nigerian classrooms and infrastructure, rather than relying on imported systems that often fail to meet local needs (Okonkwo & Ade-Ibijola, 2021).
- ❖ It starts small with the Computer Science Department but has room to grow into a university-wide platform, creating a more connected digital learning environment to improve academics.
- ❖ By building and adopting SmartLMS, the university shows commitment to modernizing learning while staying practical and cost-sensitive.

4. **For research and practice (After further improvements)**

- ❖ The project adds to the global conversation on AI in education, showing how these technologies can be applied responsibly in African contexts (Hwang et al., 2020).
- ❖ It brings local evidence to the table, since most published studies still focus on Western experiences (Goksel & Bozkurt, 2019).
- ❖ In the long run, it could serve as a model for other Nigerian and African universities looking for affordable and relevant ways to integrate AI into higher education.

1.5 SCOPE OF THE STUDY

This project looks at building and testing **SmartLMS**, a learning management system designed with the **Computer Science Department of the University of Benin** as its starting point. The idea is not to compete with or replace big international platforms but to create something practical, simple, and tailored to the real needs of students and lecturers here. By clearly setting boundaries for what this study will and will not cover, the project stays focused and realistic.

The scope is defined as follows:

- i. **Platform coverage:** SmartLMS will be developed as a web-based platform that works smoothly on both **mobile devices and desktop computers**, giving students and lecturers flexibility no matter what device, they prefer.
- ii. **Core features:** The system will provide **AI-powered support** such as practice questions generation for students, quick lecture summaries, note extraction, and student performance analytics. It will also include tools to help lecturers manage assessments, announcements and feedbacks to students more easily.

- iii. **Assessment role:** SmartLMS is not meant to replace lecturers. All **grading, assessments, uploading of materials and lecture notes and evaluation will remain fully in their hands** as the system only supports, it does not take over.
- iv. **Pilot case study:** The first version of the system will be tested within the **Computer Science Department**. Lessons learned from this pilot will guide improvements and show how the platform could later expand to other departments.
- v. **Stakeholder focus:** The project is built around two main groups: **students**, who will use the features to aid their learning and study more effectively; and **lecturers**, who will use the tools to support teaching, review AI outputs, and track student progress.
- vi. **Research boundaries:** The study will not cover a **university-wide rollout** at this stage. Instead, it will create a working prototype and pilot model that can be scaled up in the future.

1.6 LIMITATIONS OF THE STUDY

Like any academic project, this study has its own boundaries and challenges. While SmartLMS aims to make teaching and learning easier through AI support, some factors naturally limit how far the project can go at this stage. Being clear about these limitations helps set realistic expectations and guides future work.

The limitations are as follows:

- i. **Pilot size:** The system will only be tested within the **Computer Science Department**. Results may not fully represent how it would perform across all faculties or the entire university.

- ii. **Time frame:** Because this project is being carried out within the limited period of a final-year period and activities, the prototype may not include every advanced feature that could be added in the future.
- iii. **Resources:** Development is limited by available funding, infrastructure, and access to high-end servers or AI models. These constraints may affect the scale and sophistication of the platform.
- iv. **AI capability:** It is common knowledge that AI is not 100% error-free. While AI will support tasks like practice questions generation, summarization of notes and course materials, and students' performance analytics, it may not always be 100% accurate. Human oversight is required to validate and guide its use and students are advised to check back with lecturer notes for specific and sensitive topics.
- v. **Internet access:** Since the platform is web-based, its effectiveness depends heavily on stable internet connectivity, which can still be inconsistent for many students in Nigeria.
- vi. **User adoption:** Acceptance by lecturers and students may vary, and some may resist shifting to a new system due to familiarity with traditional teaching methods.
- vii. **Not a full replacement:** The system is designed to support, not replace, lecturers. Automated grading and advanced AI teaching functions are deliberately left out at this stage to maintain trust, academic authority, integrity and organization.

1.7 DEFINITION OF TERMS

- i. **Learning Management System (LMS):** An online platform that lecturers use to share course materials, assignments, and announcements, while students use it to access resources, submit work, and track their learning.

- ii. **Artificial Intelligence (AI):** Computer programs that can “think” in limited ways, such as summarizing notes, creating questions, or analyzing student performance, to make learning more interactive and personalized.
- iii. **SmartLMS:** The name of the system designed in this study. It combines the normal functions of an LMS with AI features to better support teaching and learning for both the lecturer and the student.
- iv. **Mobile-first design:** A way of creating the system so it works very well on phones first (since most students use them) but is also fully available on laptops and desktops.
- v. **Summarization:** A feature where the AI can take long lecture notes or readings and produce shorter versions that highlight the main points.
- vi. **Test and Practice Generation:** The ability of the system to automatically create practice quizzes, exercises, or mock tests for the students to help them prepare well for exams.
- vii. **Performance Analytics:** A tool in the system that tracks how students are performing over time and gives insights to both lecturers and students about progress and areas that need improvement.
- viii. **User Adoption:** The level of acceptance and willingness of students and lecturers to actually use the system once it is developed.

CHAPTER 2

Literature Review

2.1 BACKGROUND

Every academic project always comes out of an existing past research work. A literature review allows researchers to understand what others have already done in that work, the challenges that still exist in that work and where new solutions can be applied to make a difference and produce a better work. In this chapter, attention is on studies and ideas connected to Learning Management Systems (LMS), Artificial Intelligence (AI) in education, and modern smart learning environments that are currently transforming how people learn nowadays.

Over the years, education has gradually moved and evolved beyond the limits of the physical four walls of classrooms. Many educational institutions like universities, polytechnic now rely on modern digital platforms to deliver educational services to their students such as lessons, course management and communication with students. Popular platforms such as Moodle, blackboard and Canvas have made learning online flexible and accessible. Yet, most of these platforms still work mainly as administrative tools, for sharing materials, uploading assignments and grading, rather than as an active companion that truly adapts and match each student's learning pace or needs (Zawacki-Richter et al., 2019).

With the increase in Artificial Intelligence, new opportunities and means are emerging to change that reality. Ai makes it possible to create learning systems that can create practice questions, summarize course materials, track student performance and even guide students on better study habits. These new advancements are generally inspiring and provide students and teachers new learning ways that are interactive, personalized and data-driven (Wang et al., 2024).

This study builds on those existing ideas through the development of smartLMS, an improved learning management system that is designed to the familiar structure of online learning with intelligent tools to make studying more engaging and efficient.

So, this chapter explores existing theories, past research work, and current technologies in this area. It also reviews existing LMS platforms, identifies where they perform well, and where they do not, and how smartLMS aims to remedy those gaps in the context of higher education.

2.2 CONCEPTUAL REVIEW

The conceptual review gives us a clear understanding of the main ideas and terms or language that shapes this study. It explains how Learning Management Systems (LMS), Artificial Intelligence (AI), and smart learning technologies have evolved over time and how they relate to one another and work together to improve education delivery.

2.2.1 LEARNING MANAGEMENT SYSTEMS

A Learning Management System (LMS) is a digital platform created to manage and deliver educational content. It serves as a connecting bridge between teachers and students, and offers a platform for uploading course materials, submitting assessments and assignments, grading and providing feedback. Over the years, LMS platforms such as Moodle, Blackboard and Canvas have become very key tools for both in-person and online learning (Zawacki-Richter et al., 2019).

The main advantage of LMS platforms is their ability to organize learning activities efficiently. They help lecturers save time in course management and allows students to learn at their own pace. However, many traditional LMS platforms still work in a static way. They deliver information but don't really adapt to students' style of learning. This drawback has led

researchers and institutions to explore new technologies that will make LMS platforms more interactive, personalized and smart.

2.2.2 ARTIFICIAL INTELLIGENCE IN EDUCATION (AIED)

Artificial Intelligence (AI) in education focuses on using smart technologies to make teaching and learning experiences better for both the teachers and students. AI can analyze data, understand student behaviors and come up with personalized learning recommendations to improve learning. Example includes chatbots that answer student questions, system that automatically generate practice questions and tests and algorithms that track student performance (Wang et al., 2024).

AI helps teachers to make better decisions by providing helpful insights to students' progress and learning challenges. It also gives students more control over their learning with features like adaptive testing and content summarization. However, even with these opportunities, many institutions, especially in developing countries have not adopted AI fully in their learning systems, often due to technical, financial or infrastructural drawbacks.

2.2.3 SMART LEARNING SYSTEMS

Smart learning systems combine traditional LMS features with AI-powered tools to create a more responsive, interactive and engaging learning environment. These systems go beyond storing and displaying content, they have the ability to learn from user behavior, recommend personalized materials, and track performance.

According to recent studies, the idea of “smart learning” involves combining technology, analytics, and pedagogy to support both teaching and self-paced learning (Wang et al., 2024).

Systems like these have the potential to make education more flexible and general, especially in

higher institutions where students have different learning speeds and styles. The proposed smartLMS aligns with this concept by offering intelligent assistance for test generation, content summarization, and performance analytics.

2.2.4 MOBILE AND CROSS-PLATFORM LEARNING SYSTEMS

The widespread use of smartphones has reshaped how students access learning materials. Mobile learning systems allow students to study anytime and anywhere, making education more flexible and convenient. This is because, in this present age, most students are always with their mobile phones and can easily access them. However, many platforms still struggle with inconsistent mobile performance or poor user experience on smaller screens.

A cross-platform design ensures that both mobile and desktop users have equal and uninterrupted access to features and content. This flexibility is very vital in Nigerian Universities, as students rely more on mobile phones than laptops. SmartLMS is designed with this reality in mind, ensuring that its tools and features work seamlessly across devices, and it supports students and lecturers regardless of their preferred device.

2.3 THEORETICAL REVIEW

Every research work is guided by certain theories that explain how and why things work the way they do. In this section, the key learning theories that support the development of SmartLMS are discussed. These theories help to justify the design choices made in this study and show how technology can be used to improve learning and teaching outcomes.

2.3.1 CONSTRUCTIVIST LEARNING THEORY

The Constructivist Learning Theory was introduced by Jean Piaget (1964) and later improved on by other scholars. This theory emphasizes that learners build their own understanding through

experiences and active participation. It argues that learning is not just about receiving information but about constructing meaning through exploration, problem-solving, and reflection (Bada, 2015).

In relation to this study, SmartLMS is designed to create a more interactive learning experience where students can explore course materials, practice with automatically generated tests, and receive instant feedback. The AI tools integrated with the system, such as content summarization and practice question generation help to support active learning and student engagement, which are very key principles of constructivism.

2.3.2 COGNITIVE LOAD THEORY

The Cognitive Load Theory (Sweller, 1994). This theory focuses on how the human brain processes information. It suggests that when learners are overloaded with too much information at once, learning becomes less effective. So, to promote better understanding, learning and educational materials should be structured in a way that reduces unnecessary cognitive effort.

SmartLMS applies this theory by using AI to simplify complex learning materials. For example, the summarization feature can simplify lengthy and complex lecture notes into shorter, simpler and easier to understand points. Also, adaptive practice questions help students focus on key topics and test themselves rather than being overwhelmed by large amounts of content. This approach aligns with the goal of minimizing cognitive overload while maximizing understanding in students.

2.3.3 SELF-REGULATED LEARNING THEORY

Self-Regulated Learning Theory (Zimmerman, 2000). This explains how students take control of their own learning process through goal-setting, self-monitoring, and reflection. In a digital

platform, self-regulated learning becomes especially important, as students often love to learn independently at their own pace without constant supervision from lecturers.

SmartLMS supports self-regulated learning by providing students with real-time performance analytics and progress tracking. These tools help learners see and reflect on their performance, see the areas they need to improve on and adjust their study patterns accordingly. The presence of AI-generated feedback also allows students to continuously assess themselves and improve without waiting for external evaluation.

2.3.4 CONNECTIVISM THEORY

The Connectivism Theory (Siemens, 2005). This theory views learning as a process of connecting information across networks, whether through people, systems, or digital resources. It argues that in the modern digital age, knowledge is distributed across technology and social platforms, and learners gain understanding by forming meaningful connections with that knowledge.

SmartLMS reflects this idea by integrating digital collaboration features and AI tools that link students to relevant materials and resources. Through connected learning spaces and access to AI-powered summaries or practice exercises, students can connect ideas and knowledge from different sources, building a richer understanding of their courses.

Altogether, these theories provide the foundation for SmartLMS. They show that technology can support active participation, simplify learning materials, promote self-regulation, and strengthen connections and collaboration within the digital learning community.

2.4 EMPIRICAL REVIEW

Empirical studies provide practical evidence and insights into the usage of how Learning Management Systems (LMS) and Artificial Intelligence (AI) and how they have been used to improve teaching and learning. Reviewing these studies helps to identify the progress made so far, the challenges that still persist, and the gaps that this research aims to fill.

This section looks into selected international, national, and local studies related to AI-enhanced learning systems, their methods, and key findings.

2.4.1 INTERNATIONAL STUDIES

- i. Wang et al. (2024) conducted a comprehensive survey on the integration of Artificial Intelligence in Learning Management Systems. Their study highlighted how AI tools such as recommendation systems, intelligent tutoring, and automated test generation are being applied to improve learning outcomes globally. They found that while AI has improved personalization and accessibility, many systems still face challenges of data privacy, ethical concerns, and uneven lecturer involvement.

SmartLMS improves on this by addressing these issues by maintaining lecturer control over all AI-generated content, ensuring transparency, and keeping human oversight/control at the center of all assessments.

- ii. Similarly, Zawacki-Richter et al. (2019) reviewed research trends in Artificial Intelligence in higher education. They found that most AI applications focused on student support and learning analytics, but very few emphasized the teacher's perspective. The study called for more systems that empower educators rather than replace them.

But in SmartLMS, the lecturer is positioned as the supervisors of all AI outputs, and grading remains manual and solely the lecturer's job, and the AI only assists with

generation of course summary and practice review questions, analytics, and feedback. It is there to support the lecturers, not to replace them in any way.

- iii. In another study, Chen and Xie (2022) explored adaptive learning environments powered by AI algorithms. They discovered that personalized learning paths can increase student motivation and retention but require significant computational resources and reliable data networks.

But the SmartLMS system adopts a lightweight AI design suitable for platforms with limited resources like Nigerian universities, using simple text-based summaries and analytics that don't require heavy processing power.

2.4.2 NATIONAL STUDIES (NIGERIA)

- i. Adamu and Kamar (2023) made findings into the use of Moodle LMS across Nigerian universities during the COVID-19 pandemic. Their research showed that while many institutions adopted LMS platforms for remote teaching, some factors affected their usage and effectiveness. These factors include poor internet access, lack of training, and limited system flexibility.

SmartLMS is designed to be more accessible, mobile-friendly, and intuitive for both lecturers and students, making it easier for students to easily get used to using it.

- ii. In another research, Okafor and Nwosu (2022) assessed the effectiveness of digital learning tools in Nigerian tertiary institutions. They observed that most platforms lacked adaptive learning features and student analytics, leading to low engagement and poor follow-up on student progress.

To remedy this, SmartLMS integrates analytics dashboards that tracks student performance and help lecturers identify struggling students early, and also help the students see their academic strength.

2.4.3 LOCAL STUDIES (UNIVERSITY CONTEXT)

- i. A local study by Adebayo et al. (2023) examined student satisfaction with e-learning tools in selected Nigerian universities. The results showed that students appreciated flexibility but desired more personalized support and interactive features.

In SmartLMS, the inclusion of AI-generated summaries, practice questions, and performance insights directly responds to these needs, creating a more student-centered experience.

- ii. Furthermore, Olowe (2022) explored the challenges of e-learning implementation at the University of Benin. The study reported infrastructural limitations, inconsistent adoption, and lack of lecturer motivation to use LMS platforms.

But SmartLMS aims to make system management simpler for lecturers while keeping them in full control of teaching content and student assessments.

SUMMARY OF FINDINGS

From the reviewed studies, it is clear that while LMS platforms have greatly improved digital learning, several challenges remain:

- i. Limited personalization and AI integration.
- ii. Low engagement due to rigid system design.
- iii. Poor adaptability to local learning contexts.
- iv. Inadequate lecturer involvement in AI-based platforms.

SmartLMS seeks to address these gaps by creating a more flexible, locally relevant system that blends AI-assisted tools with lecturer control, mobility, and user-centered design.

2.5 REVIEW OF EXISTING LEARNING MANAGEMENT SYSTEMS

Learning Management Systems such as Moodle, Blackboard, and Canvas are popularly used right now to deliver course content, manage assignments, and support communication between students and instructors online. Over the years, these platforms have grown so much and added many useful features, but they also all share common limitations, especially when it comes to lightweight, ethically-aware AI support that still keeps instructors firmly in control. Below I summarise strengths and weaknesses of each, then compare them against the vision for SmartLMS.

2.5.1 MOODLE

i. Strength:

- ❖ Moodle is open-source and highly customizable. And because of this, many universities use it because it can be customized to local needs and budgets. It has robust tools for content delivery, quizzes, forums, and plugin ecosystem.

ii. Weakness:

- ❖ Out-of-the-box, it focuses on content/assignment management rather than intelligent personalization or real-time learning analytics (Zawacki-Richter et al., 2019).
- ❖ Requires technical know-how to configure advanced features; user experience can feel clunky on mobile unless properly optimized.

iii. Relevance to SmartLMS

- ❖ Moodle shows that adaptability and openness are valuable. SmartLMS should keep configurability but bundle easy, instructor-friendly AI features so lecturers don't need tech expertise to benefit.

2.5.2 BLACKBOARD

i. Strength:

- ❖ Blackboard has enterprise-level support, integrated tools for administration, assessment, and reporting. It is often favored by institutions needing vendor support and compliance features.

ii. Weaknesses:

- ❖ Heavier/licensed product with higher costs and slower rollout cycles for new features (many institutions find innovation lagging).
- ❖ AI features are increasingly available but can be opaque; lecturers may not have the final say on AI-generated outputs unless workflows are explicitly designed for human review.

iii. Relevance to SmartLMS:

- ❖ Blackboard demonstrates the value of institutional support and maturity. SmartLMS can borrow the usability and reporting strengths but keep the system lighter and more transparent about AI assistance.

2.5.3 CANVAS

i. Strengths:

- ❖ Canvas has modern UI, good mobile support, and an ecosystem that promotes developer integrations and third-party tools.

ii. Weaknesses:

- ❖ While Canvas offers integrations with analytics and some AI services, many institutions only use a fraction of those features. Integration complexity and cost can limit instructor uptake.

iii. Relevance to SmartLMS:

- ❖ Canvas proves that good UX and mobile readiness increase adoption. SmartLMS should match this ease-of-use while building AI tools that lecturers can accept and control.

2.5.4 EMERGING AI-ENHANCED LMS TRENDS

Recent literature shows a push toward AI features in LMS: adaptive learning paths, intelligent tutoring, chatbots, automatic feedback, and analytics dashboards (Wang et al., 2024; Abdallah et al., 2024). These features can improve student engagement and provide instructors with helpful signals, but the common issues noted across studies are:

- i. **Ethics & privacy:** Data collection and transparency are major concerns. Some people are scared to leave their online or give it up during platform signups.
- ii. **Teacher involvement:** Many AI systems risk sidelining instructors unless the design explicitly keeps them in control. Teachers need to have absolute control of the platform so that it doesn't defeat its sole purpose of education.
- iii. **Resource demands:** Heavy AI can require infrastructure and bandwidth not always available in all contexts (Wang et al., 2024).

2.6 RESEARCH GAP

From the reviewed literature, it is clear that a whole lot of existing Learning Management Systems (LMS) and Artificial Intelligence in Education (AIED) applications have contributed

significantly to digital learning. However, a number of critical gaps still exist that hinder their full potential in improving learning experience, especially within the Nigerian university context.

Most traditional LMS platforms such as Moodle, Blackboard, and Canvas focus mainly on content organization, communication, and assessment management (Zawacki-Richter et al., 2019). They often lack intelligent support features that can adapt to each student's learning needs or provide real-time performance check/response. Even the few systems that attempt to integrate AI do so in limited ways, focusing on automation rather than interactive, student-centered learning (Wang et al., 2024). This creates a technological gap where systems are efficient but not necessarily engaging or personalized.

Another major limitation identified across several studies is the overdependence on imported or foreign-built LMS frameworks. These systems are rarely optimized for local infrastructures, internet challenges, or academic cultures in developing countries. As Adamu and Kamar (2023) pointed out, many Nigerian institutions struggle with adoption due to technical complexity, unstable connectivity, and lack of user training. Therefore, due to this, there is a pressing need for an LMS model that is lightweight, adaptable, and built with the local environments in mind.

Furthermore, existing research emphasizes quantitative and experimental methods, often overlooking the qualitative insights from real user experiences, such as how lecturers and students actually interact with AI-driven tools in daily academic settings (Wang et al., 2024). This has left a practical gap in understanding how AI can support educators rather than replace them. Ethical and transparency issues, including data privacy and algorithmic bias, also remain underexplored in many AI-related studies (Akgun & Greenhow, 2022).

Finally, most available AI-integrated LMS prototypes focus heavily on automation, such as auto-grading or auto-recommendations, with little regard for lecturer oversight or human feedback.

This creates a pedagogical gap between machine efficiency and human judgment.

The SmartLMS project is designed to fill these gaps by:

- i. Integrating AI assistance features like course summary generation, practice test creation, and student performance analytics without removing the lecturer's central role of grading and overseeing everything.
- ii. Designing a locally adaptable and user-friendly system that suits both high and low-resource learning environments.
- iii. Promoting transparency and human-centered AI, ensuring that educators maintain full control over all AI-generated outputs.

In summary, this study responds directly to the technological, contextual, and pedagogical gaps identified in previous works by creating an AI-assisted LMS that aligns modern digital innovation with real human needs in education.

2.7 SUMMARY OF LITERATURE REVIEW

From the review of past studies, it is clear that Learning Management Systems (LMS) have changed the way education works today. They have made learning more flexible, accessible, and interactive for both teachers and students. Platforms like Moodle, Blackboard, and Canvas have helped schools organize courses, share learning materials, manage assignments, and allow communication between lecturers and students (Zawacki-Richter et al., 2019).

However, these systems still have several weaknesses. Many of them do not offer personalized learning experiences, and they make little or no use of Artificial Intelligence (AI) to support

students' individual learning needs. They are also usually rigid and not easily adaptable to the way students learn differently from one another.

With the growth of Artificial Intelligence in Education (AIED), there are now more chances to improve learning systems. Studies show that AI can make learning more engaging by adding features like smart feedback, automated course note summaries, adaptive learning, and students performance tracking (Wang et al., 2024). But even with these possibilities, many systems still lack proper AI integration or use it in ways that reduce the lecturer's role or sideline the lecturer. This raises concerns about data privacy, fairness, and control (Abdallah et al., 2024).

Another observation from previous research is that most studies pay more attention to the technology itself rather than the people using it, the lecturers and students (Zawacki-Richter et al., 2019). What is really needed is a system that combines AI with real human interaction so that the technology supports learning instead of taking over it. Many of the current LMS platforms also require high technical skills or strong internet access, which makes them difficult to use in places like Nigeria where digital infrastructure is still growing.

In terms of research methods, most studies rely on experiments and numbers, while very few explore the real experiences of users. Some common theories used include constructivist learning theory, learning style theory, and cognitive load theory, which all show that students learn better when the system supports their individual needs and encourages active participation (Piaget, 1964; Dantas & Cunha, 2020).

To sum it up, past research and existing systems have done a lot to improve online learning, but there are still clear gaps, such as:

- i. Lack of personalization for individual learners.

- ii. Poor integration of AI into daily academic activities.
- iii. Few ethical safeguards for AI-assisted learning.
- iv. Limited support for schools with low digital resources.
- v. Too much focus on technology features rather than the user experience.

The SmartLMS project aims to fill these gaps by building a simple, flexible, and human-centered system that uses AI to help lecturers and students work better together. It will combine smart tools like test generation, note summaries, and performance tracking, while keeping lecturers fully in control.

CHAPTER 3

METHODOLOGY AND SYSTEM ANALYSIS

3.1 BACKGROUND

This chapter explains the methods, processes, and tools used in designing and developing the SmartLMS system. It focuses on how the study was carried out from the initial stage of identifying the problem to the point of developing a working solution. The chapter also discusses the research design, the system development approach adopted, and the various stages involved in building and testing the system.

The aim is to provide a clear picture of how SmartLMS was created to meet the learning needs of students and lecturers. It outlines the techniques used in analyzing the existing system, identifying its weaknesses, and developing a more efficient and intelligent platform that integrates AI features such as test generation, content summarization, and performance analytics.

Overall, this chapter describes the entire development process, from planning to design, implementation, and testing to show how the SmartLMS was structured to solve the problems identified in the earlier chapters.

3.2 RESEARCH DESIGN

This study adopts a design and development research approach, which focuses on creating and testing a functional system that solves a real problem. The goal is not only to describe existing challenges in learning management systems but also to design a practical solution, the SmartLMS that addresses those challenges in a real academic environment.

The design and development approach were chosen because it allows for continuous improvement at different stages of system creation. It involves identifying user needs, analyzing problems in existing systems, designing the new system, implementing it, and then testing it to ensure that it works well. This approach makes it possible to combine both research and technical development in a single structured process.

The study was carried out within the Computer Science Department of the University of Benin, which serves as the case study. Data for the design were gathered through observation, informal discussions with students and lecturers, and a review of existing LMS platforms such as Moodle and Canvas. These findings helped to shape the requirements and features of the proposed system.

In summary, the research design follows a systematic process that moves from problem identification to solution implementation. It combines theoretical understanding with hands-on development to produce a system that can be tested, refined, and later scaled across the university.

3.3 SYSTEM DEVELOPMENT METHODOLOGY

The development of SmartLMS followed the Agile Software Development Methodology, which was chosen because of its flexibility and ability to handle changes effectively during the development process. Agile focuses on building the system in small, manageable stages called iterations or sprints, allowing continuous testing, feedback, and improvement at each stage. This approach was most suitable for SmartLMS because the system required close interaction with potential users which are the students and lecturers, to understand their real needs and challenges. By adopting Agile, it became easier to adjust the system design, add new features, and fix issues quickly based on user needs and feedback.

The Agile process used in this study followed these main stages:

1. Planning and Requirement Gathering:

At this stage, the main problems with existing LMS platforms were identified through observation and discussions with lecturers and students. The key system features and goals were defined based on these findings.

2. System Design:

This stage describes the structure of the system, including the interface layout, database design, and overall architecture. Diagrams such as use case diagrams and entity-relationship diagrams (ERD) were developed to visualize the system.

3. Implementation:

The system was developed using modern web technologies. Node.js under the ExpressJs framework was for the backend, MongoDB was used for the database, and NextJs was used for the frontend. AI features for course content summarization, practice test generation, and performance analytics were integrated.

4. Testing and Evaluation:

Each component of the system was tested to ensure it worked correctly and met user requirements. User feedback was collected during testing to improve the interface and functionality.

5. Deployment and Maintenance:

After testing, the system was prepared for deployment. Future updates and maintenance were also planned to ensure continuous improvement.

The Agile methodology allowed for an interactive and user-focused development process, ensuring that SmartLMS was not just functional but also aligned with the needs of its intended users.

3.4 SYSTEM ANALYSIS

System analysis refers to studying the existing methods of managing learning activities, identifying the challenges faced by both lecturers and students, and defining how the new system, SmartLMS can solve those problems effectively. This stage provides a detailed understanding of the current situation and forms the foundation for the system design and implementation. The analysis covers the following key areas:

3.4.1 EXISTING SYSTEM ANALYSIS

Currently, departments in the University of Benin do not have a fully functional Learning Management System (LMS) tailored to its specific needs. Most course materials are shared informally through platforms such as WhatsApp, Google Drive, or personal emails. Class announcements and assignments are often handled manually, mostly shared through the course rep, which can lead to poor organization and inconsistent access to information.

Lecturers who wish to use existing LMS platforms like Moodle or Canvas often face setup difficulties, and many of these systems require stable internet connections and high technical knowledge. As a result, their use remains very limited among both lecturers and students.

3.4.2 PROBLEMS OF THE EXISTING SYSTEM

From the review and observation, the following major problems were identified in the current system:

- i. Learning resources are scattered across multiple platforms, making it hard for students to access materials easily.
- ii. There is no unified platform for managing courses, assignments, and communication.
- iii. Limited or no AI support to help personalize learning or assist lecturers in grading and tracking student performance.
- iv. Poor student engagement due to the lack of interactive and organized learning tools.
- v. Difficulty in monitoring students' academic progress and performance trends.
- vi. Lack of proper integration of mobile-friendly features for students who primarily use their smartphones.

3.4.3 PROPOSED SYSTEM ANALYSIS

The proposed SmartLMS is designed to address these challenges by providing a centralized, intelligent, and user-friendly platform for both lecturers and students. It will combine traditional learning management features with artificial intelligence tools to create a smarter and more interactive learning experience. Key features of SmartLMS include:

- i. AI-assisted test and practice question generation.
- ii. Automatic course content summarization and key point extraction from uploaded materials.
- iii. Performance analytics to help lecturers monitor students' progress in assessments.
- iv. Course and material management for easy upload, access, and tracking.
- v. Communication tools for announcements and feedback between lecturers and students.
- vi. Accessibility on both mobile and desktop devices for convenience.

The system ensures that lecturers have full control over assessments and grading while AI features act as supportive tools to improve efficiency and learning engagement.

3.4.4 SYSTEM REQUIREMENTS

The system requirements are divided into functional and non-functional aspects:

a) Functional Requirements: This defines the specific features and operations the smartLMS system must be able to perform to meet user needs. These requirements speak more about the logical, functional and behavioral side of smartLMS. They include the following:

- i. User authentication and role management (lecturer/student).
- ii. Uploading and downloading of course materials.
- iii. Assessing external links sent by course lecturers.
- iv. AI-powered generation of practice questions and notes.
- v. Course creation and management by lecturers.
- vi. Viewing of announcements and feedback.
- vii. Student performance tracking and analytics.
- viii. Secure grading and result management by lecturers.

b) Non-Functional Requirements: These requirements define how the system should perform and operate to give the best results. They include the following:

- i. **Usability:** The system will have simple and intuitive interface for easy navigation.
- ii. **Reliability:** The system will function consistently and efficiently with minimal downtime.
- iii. **Scalability:** The system will be able to expand to other departments.
- iv. **Security:** User data will be protected through authentication and encryption.
- v. **Compatibility:** Accessibility on both desktop and mobile devices.

- vi. **Performance:** The system will have a fast response time and smooth-running operation under normal load.

In summary, the system analysis phase identifies the gaps in the existing methods of managing learning activities and establishes the foundation for building SmartLMS, a modern, AI-supported platform that enhances both teaching and learning experiences in educational institutions.

3.5 SYSTEM DESIGN

The system design phase looks at how the SmartLMS system is structured, both in terms of function and architecture. It defines how different components of the system interact, how data flows within it, and how users (students and lecturers) perform their activities.

The main goal of this stage is to convert the system requirements identified earlier into a practical blueprint that guides implementation. It includes the design of the user interface, the database, and system components that make SmartLMS functional and easy to use.

3.5.1 SYSTEM ARCHITECTURE

The SmartLMS follows a three-tier architecture, which includes the presentation layer, application layer, and database layer.

1. **Presentation Layer:** This is the user interface where lecturers and students interact with the system through web browsers or mobile devices. It is designed using Next.js to ensure a responsive and modern interface that works smoothly on both desktop and mobile devices.
2. **Application Layer:** This layer contains the system's core logic. It is built using Node.js and Express.js. It handles operations such as users authentication, course management,

AI-assisted features (like test generation and summaries), and communication between users and the database.

- 3. Database Layer:** The database is built using MongoDB, a NoSQL database that allows flexible data storage and retrieval. It stores user accounts, course materials, assessments, analytics data, and AI-generated outputs.

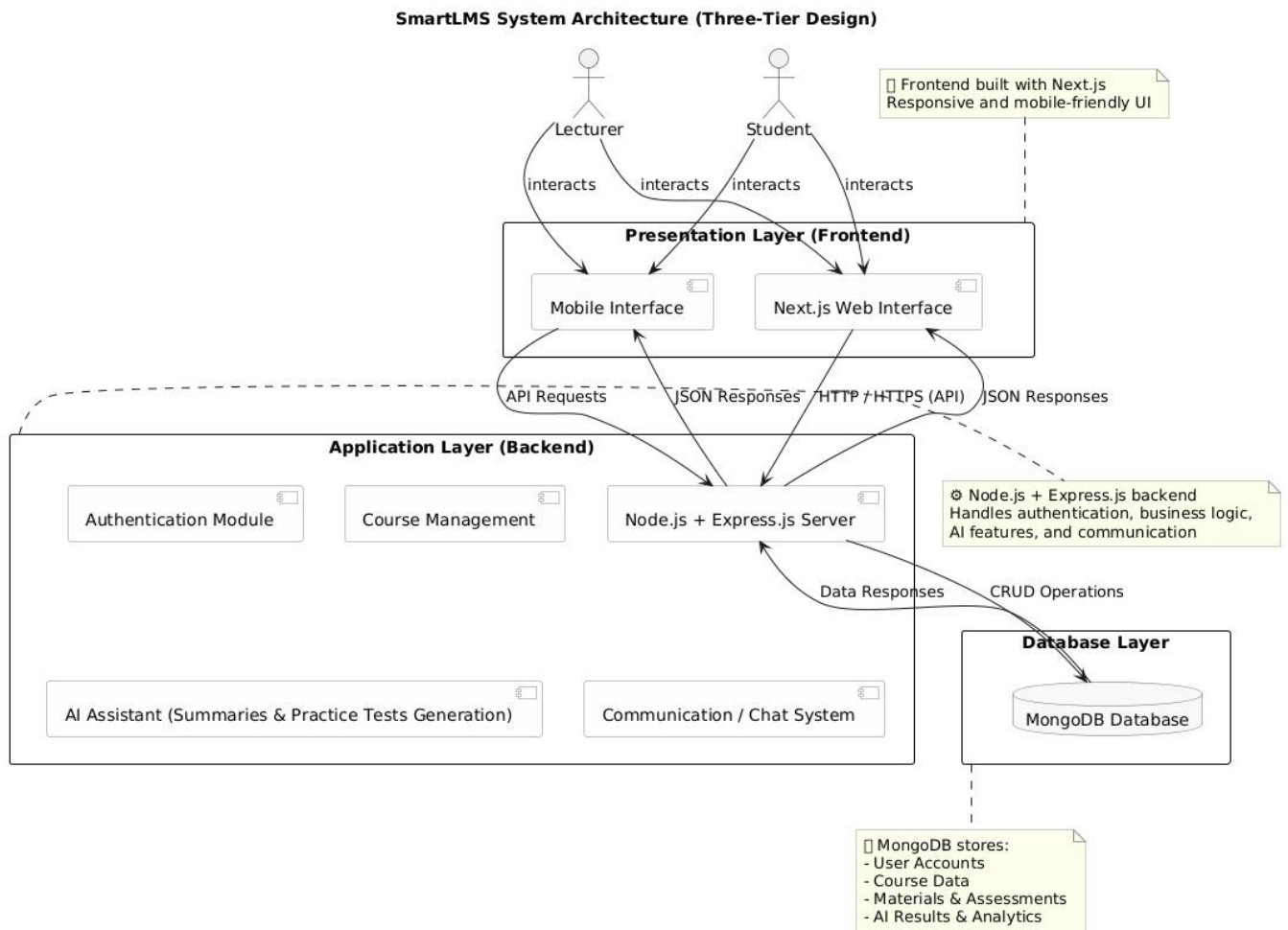


Fig 3.1 SmartLMS System Architecture (Three-Tier Design)

3.5.2 USE CASE DIAGRAM

The use case diagram shows how different users interact with the system.

Main Actors:

- i. **Lecturer**
- ii. **Student**

Use Cases Include:

- i. Lecturer logs in and manages courses (Creation and updating).
- ii. Lecturer uploads course materials.
- iii. Lecturer views student performance analytics.
- iv. Lecturer sets course assessments for students (assignments, classwork, tests).
- v. Lecturer sends announcements and feedbacks to students.
- vi. Student registers and views enrolled courses.
- vii. Student accesses materials and submits assignments.
- viii. Student generates AI practice questions.
- ix. Students generate course summaries using AI
- x. Student views performance reports and announcements from lecturers.

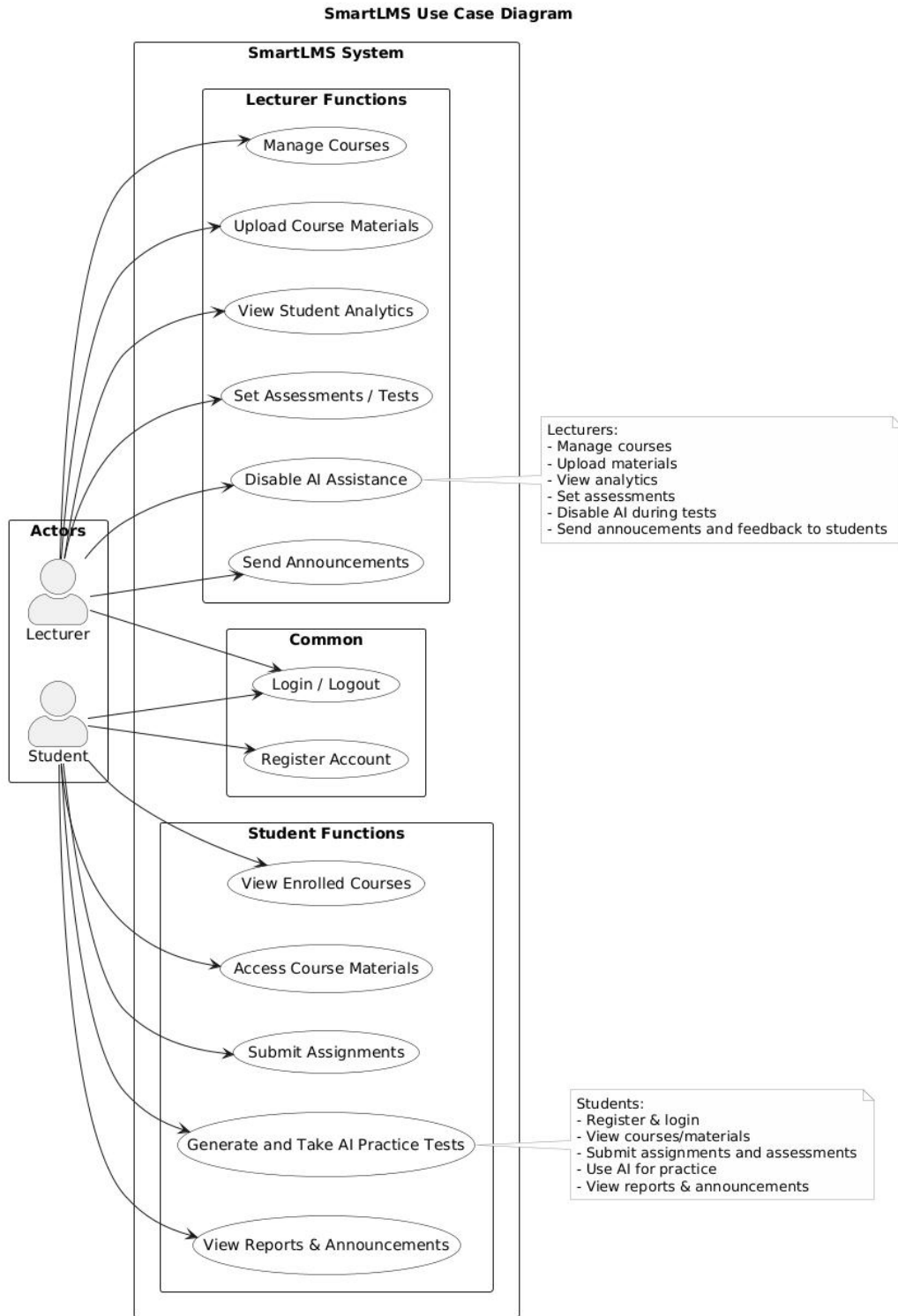


Fig 3.2 SmartLMS Use Case Diagram

3.5.3 ENTITY RELATIONSHIP DIAGRAM (ERD)

The ERD represents the database structure and how entities interact with each other.

Main Entities:

- i. **User (Lecturer/Student):** contains personal and login information.
- ii. **Course:** stores course details created by lecturers.
- iii. **Material:** represents lecture notes, videos, documents uploaded (pdf, word, excel, PowerPoint), images, links.
- iv. **Assessment:** includes tests, assignments, and quizzes.
- v. **Result:** records grades and analytics data for performance tracking.
- vi. **AI Assistant:** handles generated content such as summaries and practice questions.

Relationships:

- i. A **Lecturer** can create many **Courses**.
- ii. A **Course** can have many lecturers.
- iii. A **Course** can have multiple **Materials** and **Assessments**.
- iv. A **Student** can enroll in many **Courses** and attempt multiple **Assessments**.
- v. **Results** belong to both a **Student** and a **Course**.

SmartLMS Entity Relationship Diagram (ERD Diagram)

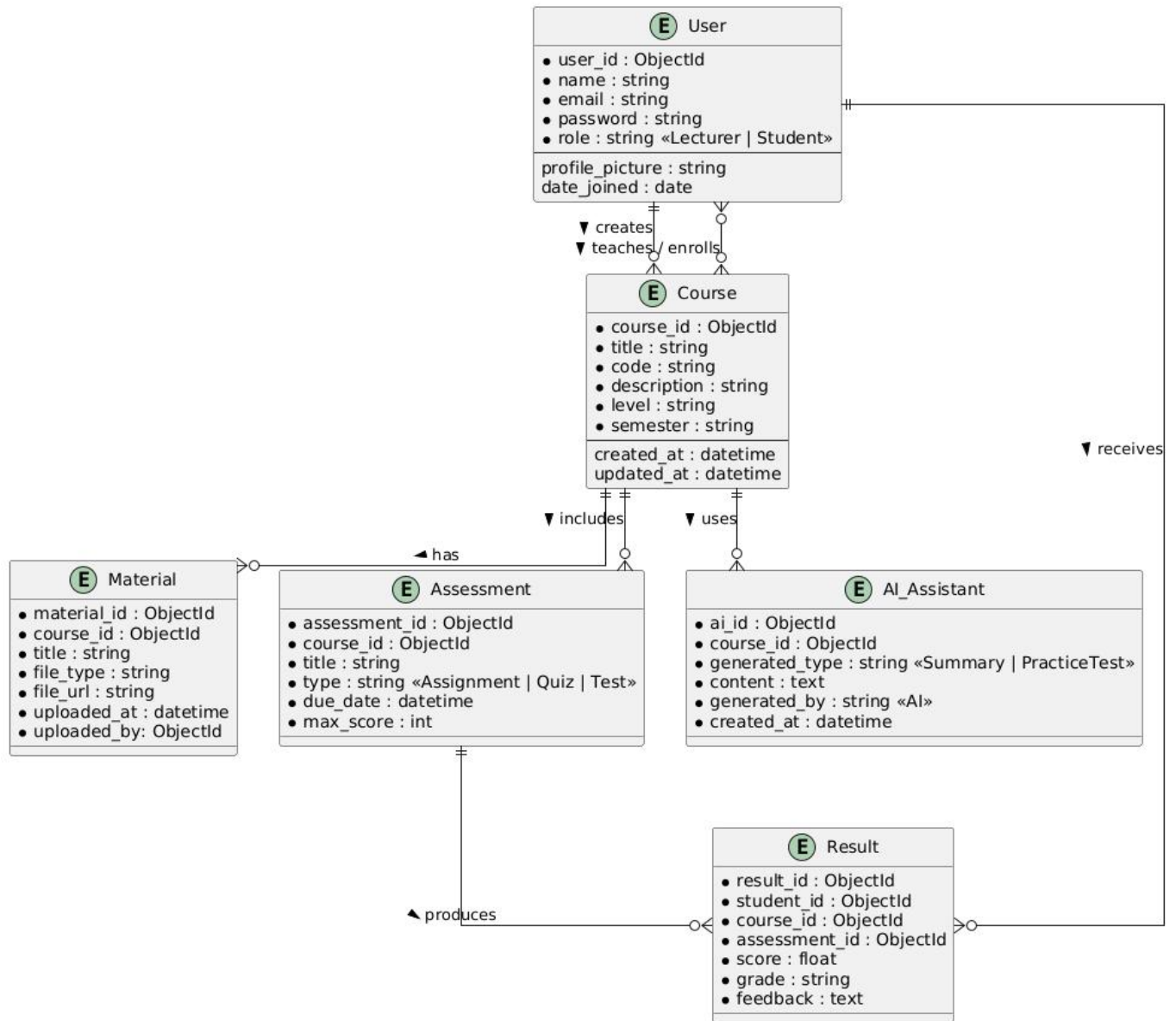


Fig 3.3 SmartLMS Entity Relationship Diagram (ERD Diagram)

3.5.3 USER INTERFACE DESIGN

The SmartLMS interface was designed with **simplicity and clarity** in mind.

Key design principles include:

- i. Clean dashboard layout for both lecturers and students.
- ii. Quick access to key features such as courses, materials, and tests.
- iii. Notifications and announcements for easy communication.
- iv. Mobile-responsive pages for convenient use on phones and tablets.

Example UI pages include:

- i. **Login/Signup Page**
- ii. **Lecturer Dashboard** (course management, analytics, AI tools)
- iii. **Student Dashboard** (course list, tests, results)
- iv. **Course Page** (materials, announcements, and activities)

In summary, the system design provides the structural and visual foundation for SmartLMS. It makes sure that all components, from database to user interface works well together to deliver an intelligent, user-friendly, and efficient platform that supports both teaching and learning.

CHAPTER 4

SYSTEM IMPLEMENTATION

4.1 BACKGROUND

This chapter focuses on the practical implementation of the SmartLMS system. It turns the concepts and designs from previous chapters into a functional system. It describes the implementation approach, the technologies and tools used, the testing strategies applied, and the evaluation of the system's effectiveness.

The main goal of this chapter is to show how the SmartLMS platform was created to meet the study's objectives. These include improving learning efficiency, providing AI-assisted features, and creating a user-friendly environment for instructors and students.

The implementation approach involves setting up the frontend, backend, and database. It also includes adding the Artificial Intelligence (AI) module that generates summaries, practice exams, and performance analytics. The testing and evaluation sections confirm that the system works as expected, meets user needs, and performs well in real-world situations.

By the end of this chapter, the full capabilities of SmartLMS will be presented. This will include screenshots, explanations of key components, and results from system testing and user feedback. This will demonstrate that the platform meets the project's aim of improving digital learning and academic engagement.

4.2 SYSTEM IMPLEMENTATION

The SmartLMS system implementation represents the phase where the design and specifications from Chapter Three became a fully functional application. This section covers the practical side

of developing the system, the tools and frameworks used, and how the different modules were combined to create a smooth learning experience for both lecturers and students.

SmartLMS was created as a web application that can be accessed on both desktop and mobile devices. The system used modern web technologies to ensure it is scalable, responsive, and easy to use. Its structure follows a three-tier setup that includes the frontend (presentation layer), backend (application layer), and database (data layer), with Artificial Intelligence (AI) capabilities added through secure APIs.

4.2.1 FRONTEND IMPLEMENTATION

The **frontend** of SmartLMS serves as the main interface where users interact with the system. It was developed using JavaScript and its framework, **Next.js**, **TailwindCSS for the styling and other frontend development libraries**, which provided flexibility, performance optimization, and responsiveness across devices.

The interface was designed with a **user-centered approach**, focusing on simplicity, clarity, and accessibility. Key pages include:

- i. **Registration and Login Pages:** For account creation and proper authenticated login.
- ii. **Dashboard:** Provides an overview of activities, available courses, announcements and some other features.
- iii. **Course Page:** Provides an overview of each course, showing the details, uploaded materials, assignments, and AI-generated study aids.
- iv. **Assessment Page:** Allows students to take assessments or view assigned tests.
- v. **AI Tools Page:** Enables students to generate note summaries or practice questions using AI.

- vi. **Feedback Page:** Enables students to view feedbacks from lecturers.

The frontend communicates with the backend through **RESTful APIs**, ensuring smooth data exchange between the client and server.

4.2.2 BACKEND IMPLEMENTATION

The backend of SmartLMS was implemented using Node.js with the Express.js framework. This layer is responsible for handling the core logic, managing user registration and authentication, storing and retrieving data, and connecting with the AI module and the database.

Key backend functions include:

- i. Managing user accounts and authentication using JWT (JSON Web Token).
- ii. Handling CRUD (Create, Read, Update, Delete) operations for courses, assessments, materials, feedbacks and user profiles.
- iii. Routing API requests between the frontend and database.
- iv. Integrating AI features through API calls for course content summarization and performance analytics.
- v. Ensuring data security, scalability, and efficient system performance.

The backend is modularized into routes, controllers, and middleware to improve maintainability and ensure clean separation of responsibilities.

4.2.3 DATABASE IMPLEMENTATION

The database was implemented using MongoDB, a NoSQL database system chosen for its flexibility, scalability, and ease of integration with Node.js.

MongoDB stores information in collections such as:

- i. **Users:** This collection stores registered users with their roles as lecturers and students.
- ii. **Courses:** This collection stores courses and their details such as title, code, course assessments, uploaded course materials, etc.
- iii. **Materials:** This store uploaded lecture files, materials and resources.
- iv. **Assessments:** This stores tests, assignments, and quizzes.
- v. **Results:** This stores grades and performance data of students.

Each item in every collection is assigned a unique id for identification.

4.2.4 AI INTEGRATION

The AI module was connected to SmartLMS via secure API endpoints such as OpenAI and Gemini AI. This integration enables the system to perform various intelligent functions such as course materials summarization, generation of self-practice test questions for students and offering feedbacks on learning process.

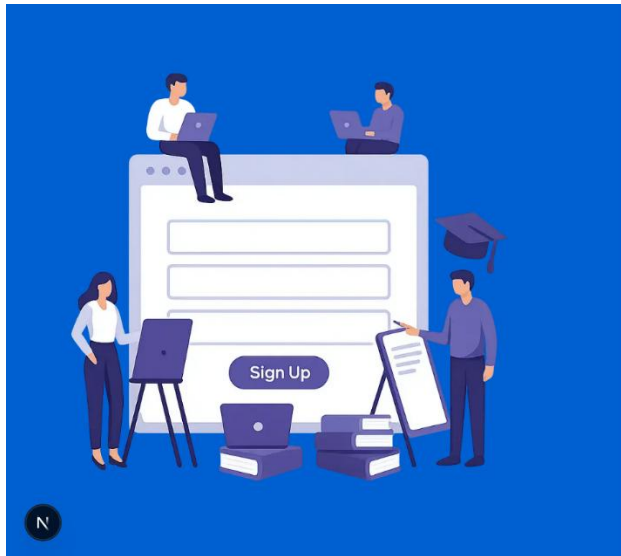
4.2.5 HOSTING AND DEPLOYMENT

SmartLMS was deployed on a cloud-based hosting platforms for accessibility and scalability.

- i. The Frontend was hosted on Vercel, which supports Next.js applications and provides continuous integrations for fast updates.
- ii. The Backend was hosted on Render, offering reliable server performances and API handling.
- iii. Cloudinary was used for storing uploaded courses materials such as course files (pdf, docx, excel), course videos and images

With this deployment setup, SmartLMS can be accessed through any standard web browser on both mobile and desktop devices, ensuring availability and convenience for all users.

4.3 SCREENSHOTS OF THE SOFTWARE



Create an account

Name

Email

Password

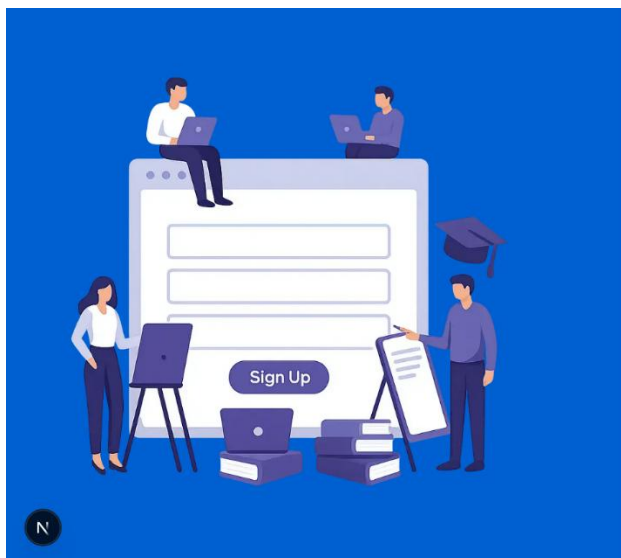
Confirm Password

Role:

[Already have an account? Login](#)

Fig 4.1 Registration page

As illustrated in Fig 4.1, this is the registration page where users (lecturers, students) register and sign up in the software. Each user sign-up with their full name, email, password and specify their role, either “lecturer” or “student”.



Login

Email

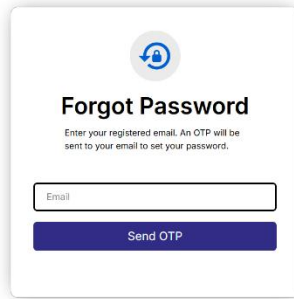
Password

[Forgot Password](#)

[Don't have an account yet? No worries! Register Here](#)

Fig 4.2 Login page

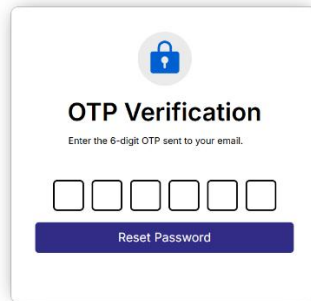
Fig 4.2 shows the login page of the software. After a successful registration, users are redirected to the login page to login with their email and password.



The 'Forgot Password' page features a circular refresh icon at the top. Below it, the title 'Forgot Password' is displayed in bold. A sub-header reads: 'Enter your registered email. An OTP will be sent to your email to set your password.' There is a text input field labeled 'Email' and a dark blue button labeled 'Send OTP'.

N

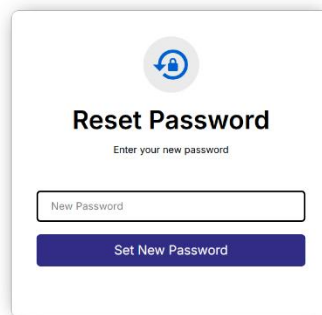
Fig 4.3 Forgot Password Page



The 'OTP Verification' page features a circular lock icon at the top. Below it, the title 'OTP Verification' is displayed in bold. A sub-header reads: 'Enter the 6-digit OTP sent to your email.' There are six empty square input boxes for the OTP digits and a dark blue button labeled 'Reset Password'.

N

Fig 4.4 OTP Verification Page



The 'Reset Password' page features a circular refresh icon at the top. Below it, the title 'Reset Password' is displayed in bold. A sub-header reads: 'Enter your new password'. There is a text input field labeled 'New Password' and a dark blue button labeled 'Set New Password'.

N

Fig 4.5 Set New Password Page

Fig 4.3, Fig 4.4 and Fig 4.5 illustrates the Forgot Password, OTP Verification Page and Set New Password page. When a user forgets his/her password, he can reset it successfully using the following pages.

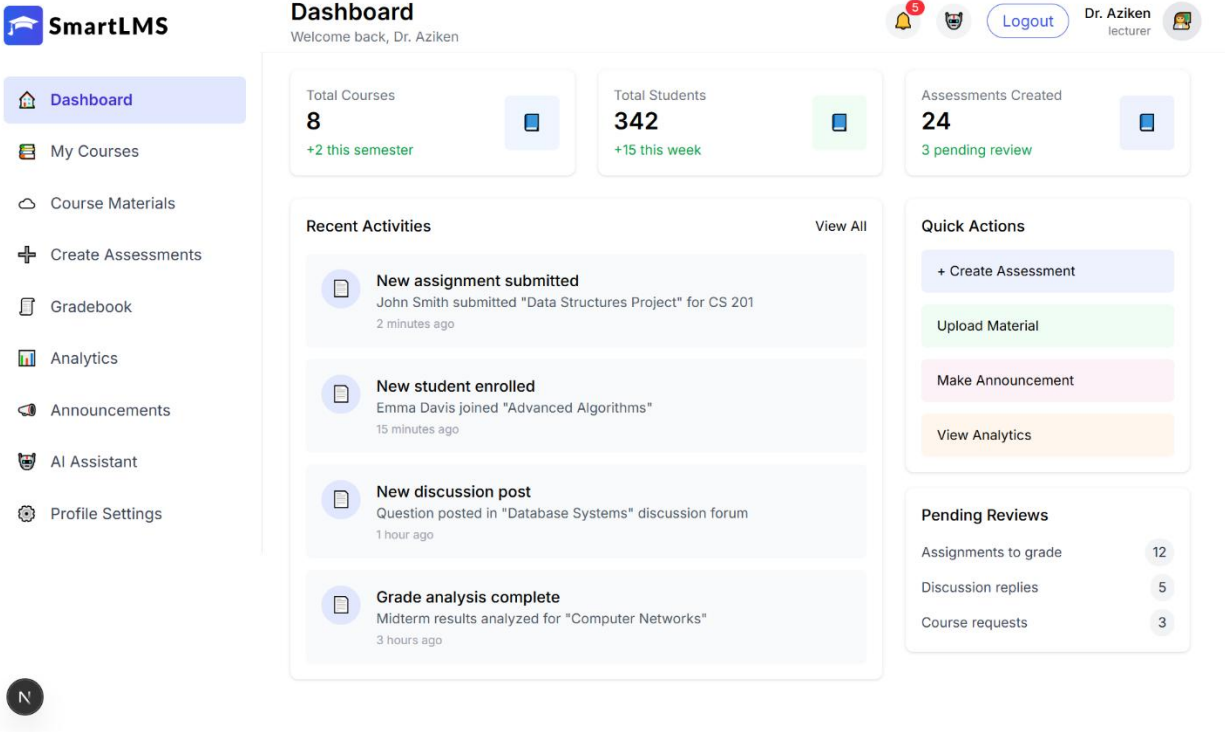


Fig 4.6 Lecturer Homepage Page

Fig 4.6 shows the lecturer home page. This is an overview of what goes on in the lecturer profile.

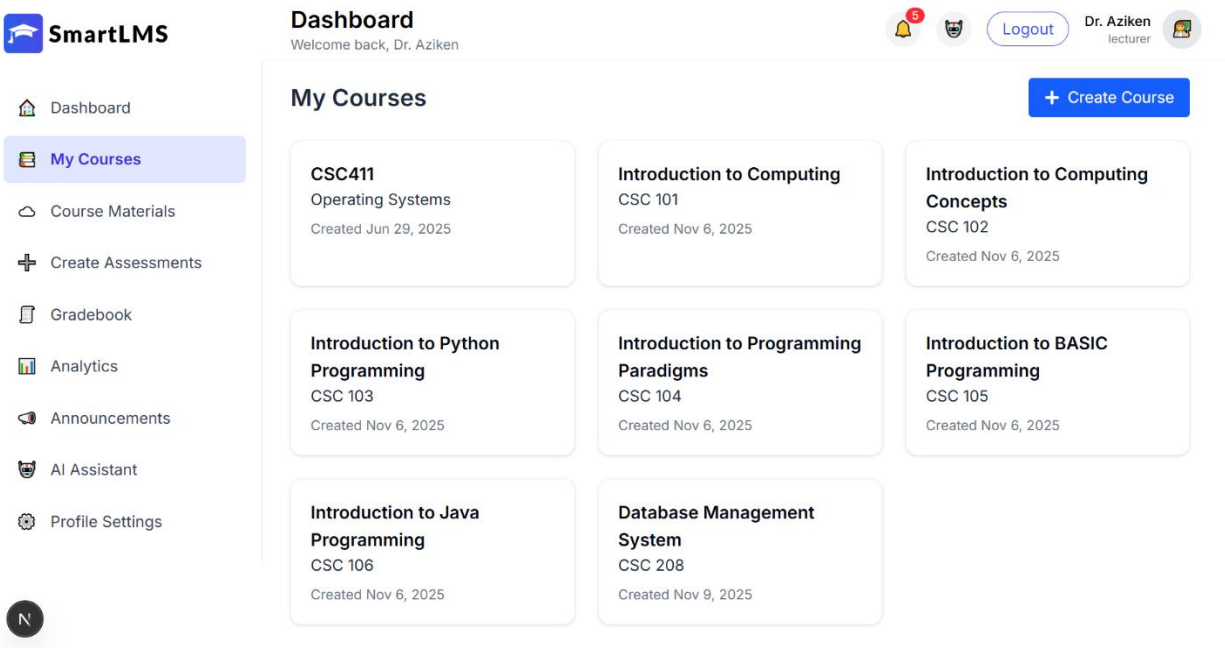


Fig 4.7 My Courses Page Shows Courses Created by Lecturer

Fig 4.7 above illustrates the page called “My Courses”. It shows all the courses created by the lecturer. It fetches all the courses directly from the database using an API call.

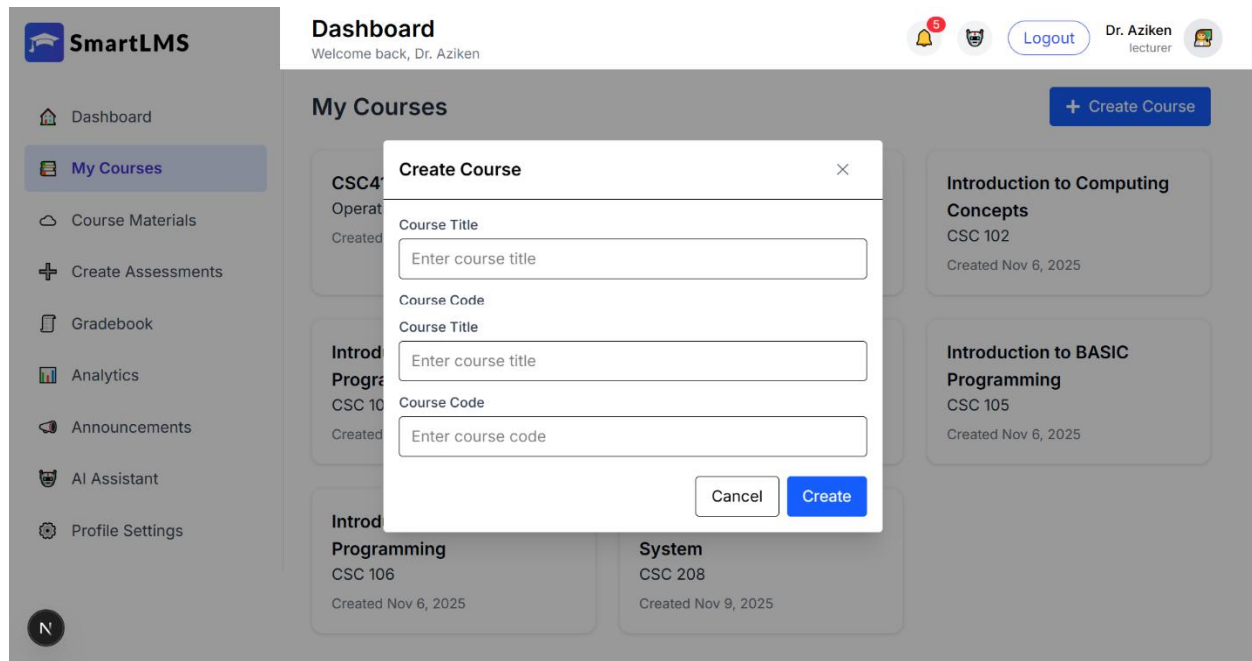


Fig 4.8 Lecturer creating a new course

As a follow-up from Fig 4.7, Fig 4.8 illustrates the page where a course can be created. When the “Create Course” button in Fig 4.7 is clicked, it opens up a pop-up that the lecturer enters all the details needed to create the course and when the “create” button is clicked, the course is successfully created and added to the database collection. After successful creation, an API call is made to the database again to re-fetch the courses collection and re-render the courses. This ensures that there are no inconsistencies between the server and the frontend.

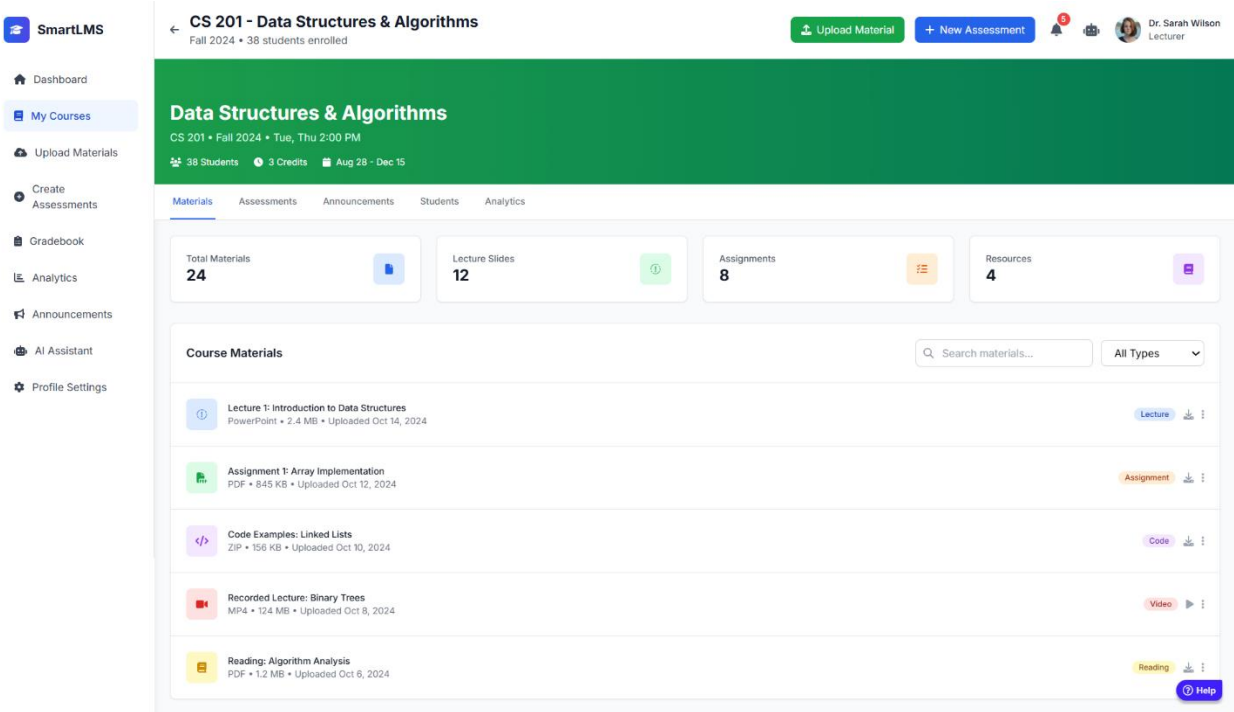


Fig 4.9 The course details page showing information about each course. The materials, assessments and announcements

Fig 4.9 shows the course details of each course has a schema structure that includes all the details concerning the course. These details include, the materials uploaded for that course, the assessment created for that course, the announcement sent for that course, etc. All these are fetched and displayed in the course details page.

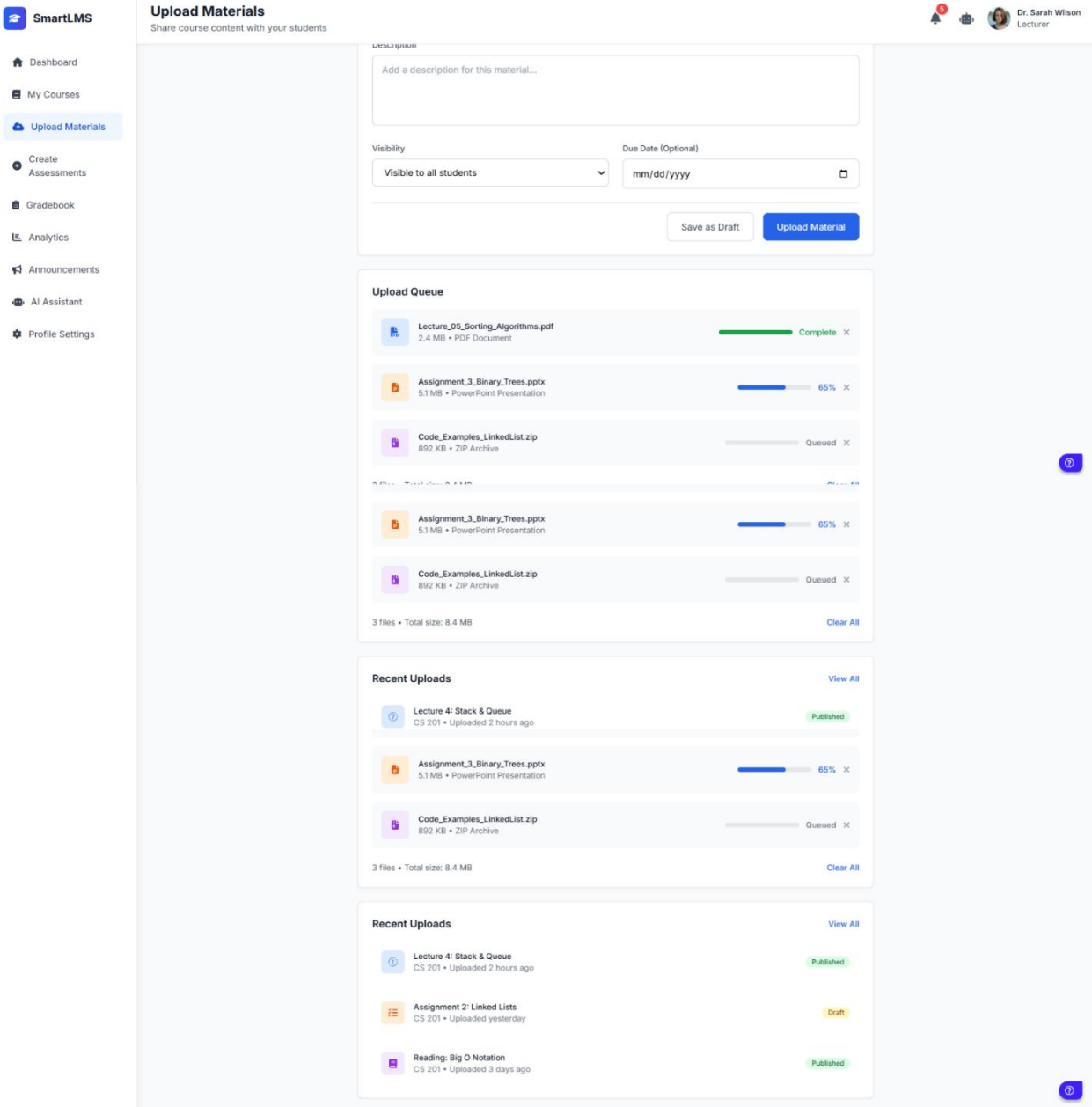


Fig 4.10 Uploading materials (different file formats) for students

Fig 4.10 illustrates materials upload for each course. This allows the lecturer to upload a material for any course. To complete this process, there are fields that need to be filled, such as material title, description, type (image, document, video, slides), etc. This page also shows the materials that has previously been uploaded and the ones undergoing upload. All uploaded materials are recorded in the database and saved in an online platform called Cloundinary.

Fig 4.11 Create an assessment (Quiz, tests, CA) for students

Fig 4.11 illustrates the Assessment Feature of smartLMS. Here the lecturer can create assessments for the students. Assessment can be on different kinds such Quiz, test, CA and kinds (theory, MCQ, true/false). The lecturer has complete control over how the assessment will look like and how it will be set. All created and published assessments are recorded and saved to the

assessments collection in the database and each can be referenced by a unique id along with each question.

The screenshot displays the SmartLMS Gradebook interface. On the left is a navigation sidebar with options like Dashboard, My Courses, Upload Materials, Create Assessments, Gradebook (highlighted), Analytics, Announcements, AI Assistant, and Profile Settings. The main content area is titled 'Gradebook' with the subtitle 'Review and grade student submissions'. At the top right, there are notification icons and the user's name 'Dr. Sarah Wilson, Lecturer'. Below the title is a 'Filters' section with dropdown menus for Course (CS 201 - Data Structures & Algr), Assessment Type (Quiz), and Status (Graded), along with a search box for students. A summary row shows: Total Submissions (142), Pending Review (18), Average Score (85.4%), and Late Submissions (7). The 'Student Submissions' table lists individual entries with columns for Student Name, Assessment, Submission Date, Score, Status, and Actions. The table shows five rows of data with various statuses like 'Graded', 'Pending Review', 'Late Submission', and 'Not Submitted'. At the bottom, there is a pagination control showing 'Showing 1 to 5 of 142 results' and a 'Help' button.

STUDENT NAME	ASSESSMENT	SUBMISSION DATE	SCORE	STATUS	ACTIONS
John Smith CS201-001	Midterm Exam - Data Structures CS 201	Oct 15, 2024 2:30 PM	92/100 92%	Graded	View Edit
Emily Johnson CS201-001	Quiz 3 - Algorithms CS 201	Oct 14, 2024 11:45 AM	-/25 Pending	Pending Review	Grade View
Michael Brown CS301-002	Assignment 2 - Database Design CS 301	Oct 16, 2024 11:59 PM	78/100 78%	Late Submission	View Edit
Sarah Davis CS401-001	Project Proposal CS 401	Oct 13, 2024 3:15 PM	95/100 95%	Graded	View Edit
David Wilson CS201-001	Quiz 3 - Algorithms CS 201	Not submitted	0/25 0%	Not Submitted	View Contact

Fig 4.12 Grading Page for published assessments

Fig 4.12 shows the grading page for assessments. Here each, each assessment answer submitted by each student is graded. There are also tracking features that allows the lecturer to see the assessments that has been graded and hasn't been graded.

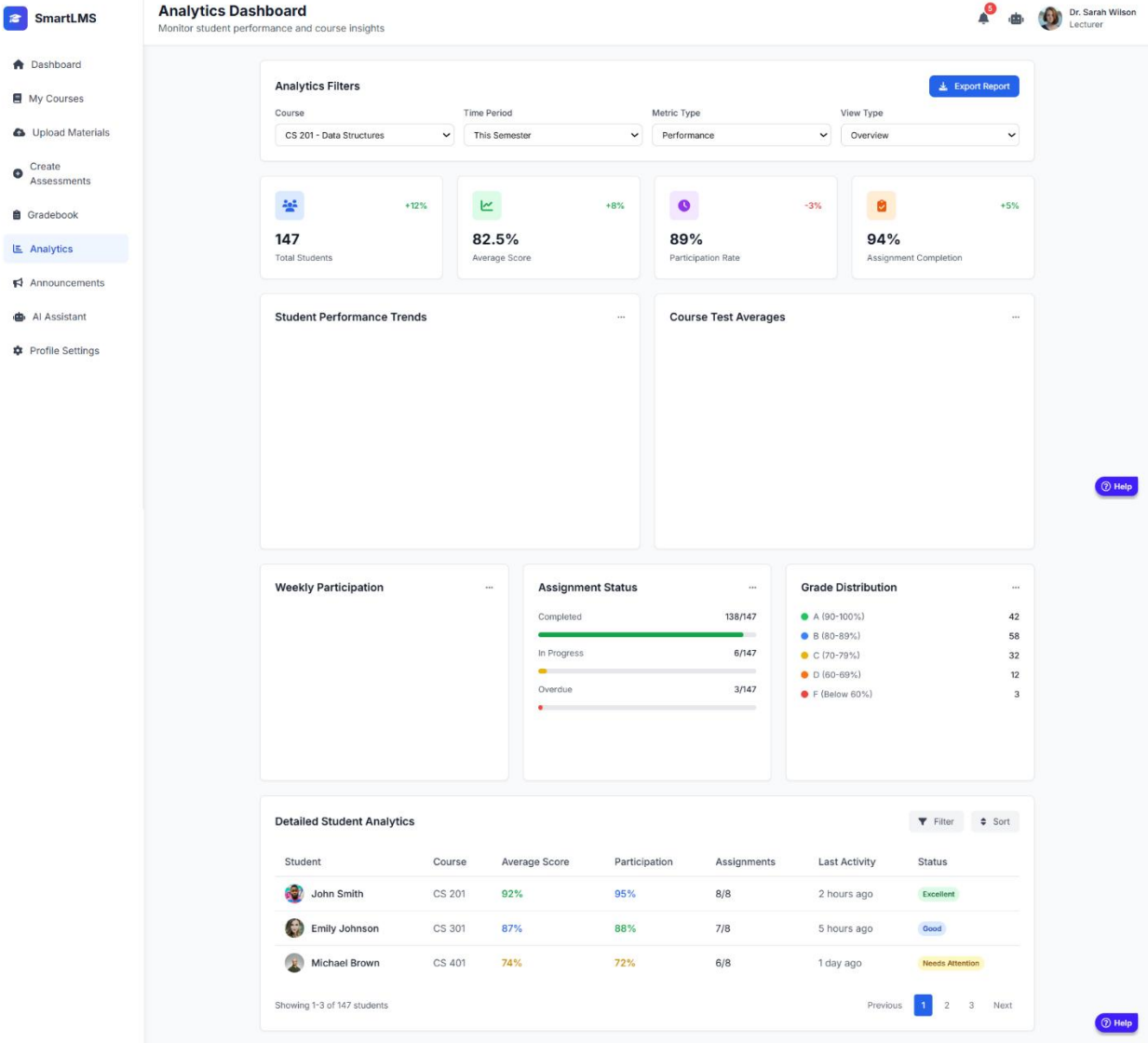


Fig 4.13 Analytics page where lecturer tracks the performance of students for each assessment and can export the analytics report in different format

Fig 4.13 illustrates the Analytics page which shows the performance of students in a course assessment. This performance analytic can also be exported in a document format.

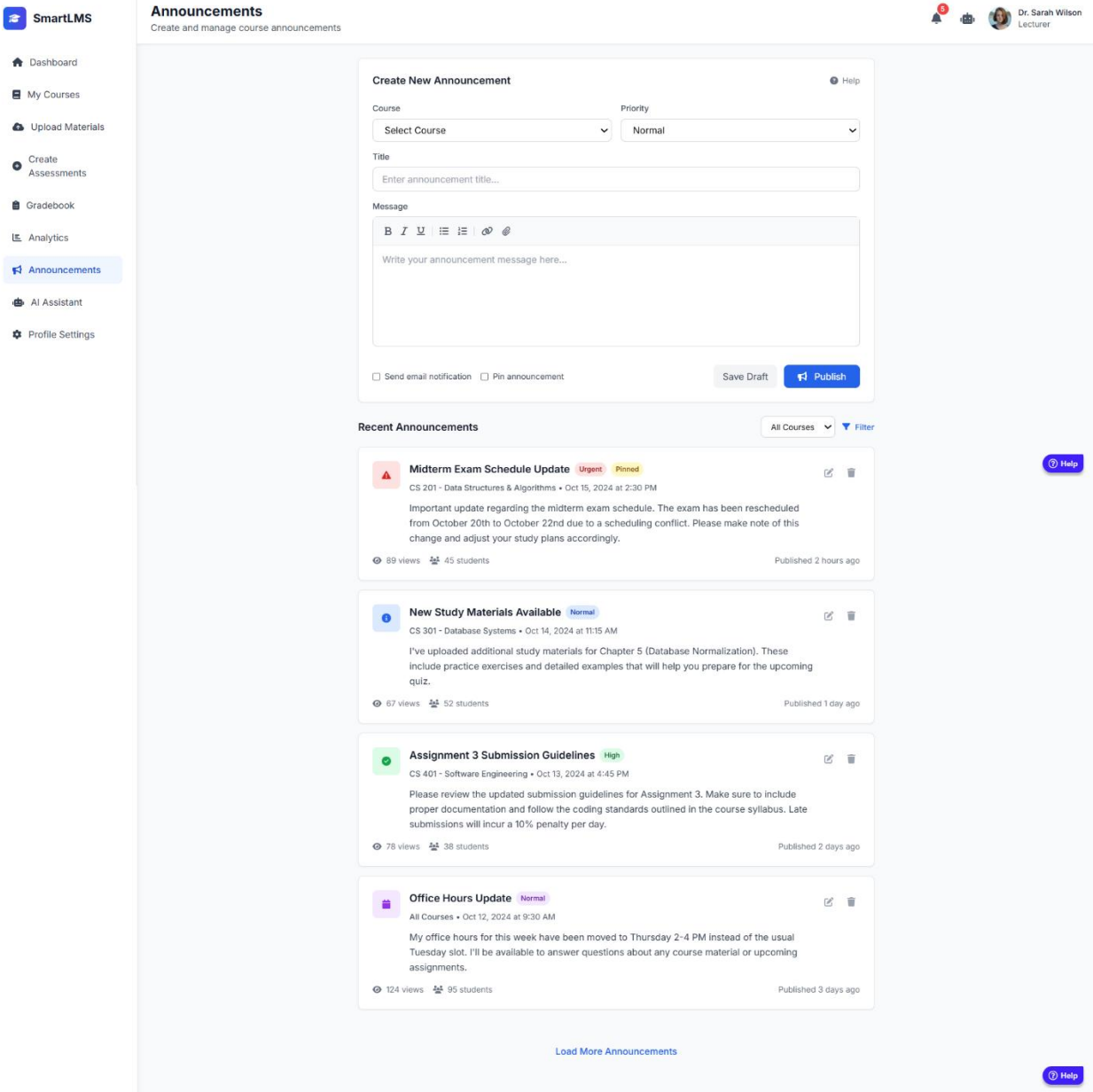


Fig 4.14 Page for Lecturer create announcements for students

Fig 4.14 illustrates the announcement page where lecturer can send announcement and vital information about a course for the students to access. Each announcement have a title and description and they are given an urgency level.

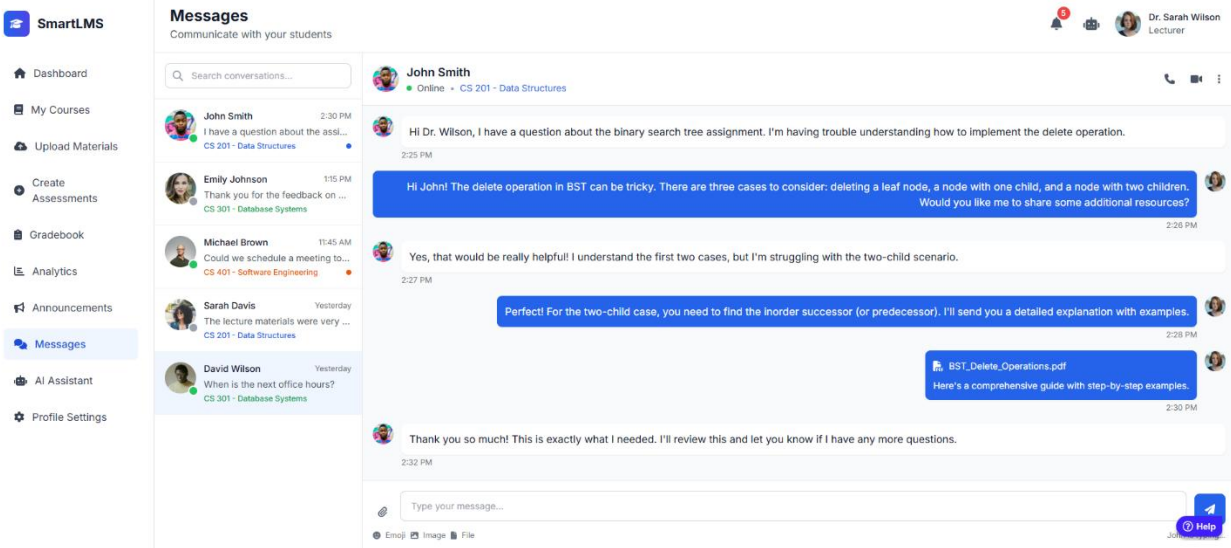


Fig 4.15 Messaging and Feedback page for easy communication between lecturers and students

Fig 4.15 illustrates the messaging/feedback page. Here, lecturer can communicate with individual student privately giving feedbacks and answers to question when needed.

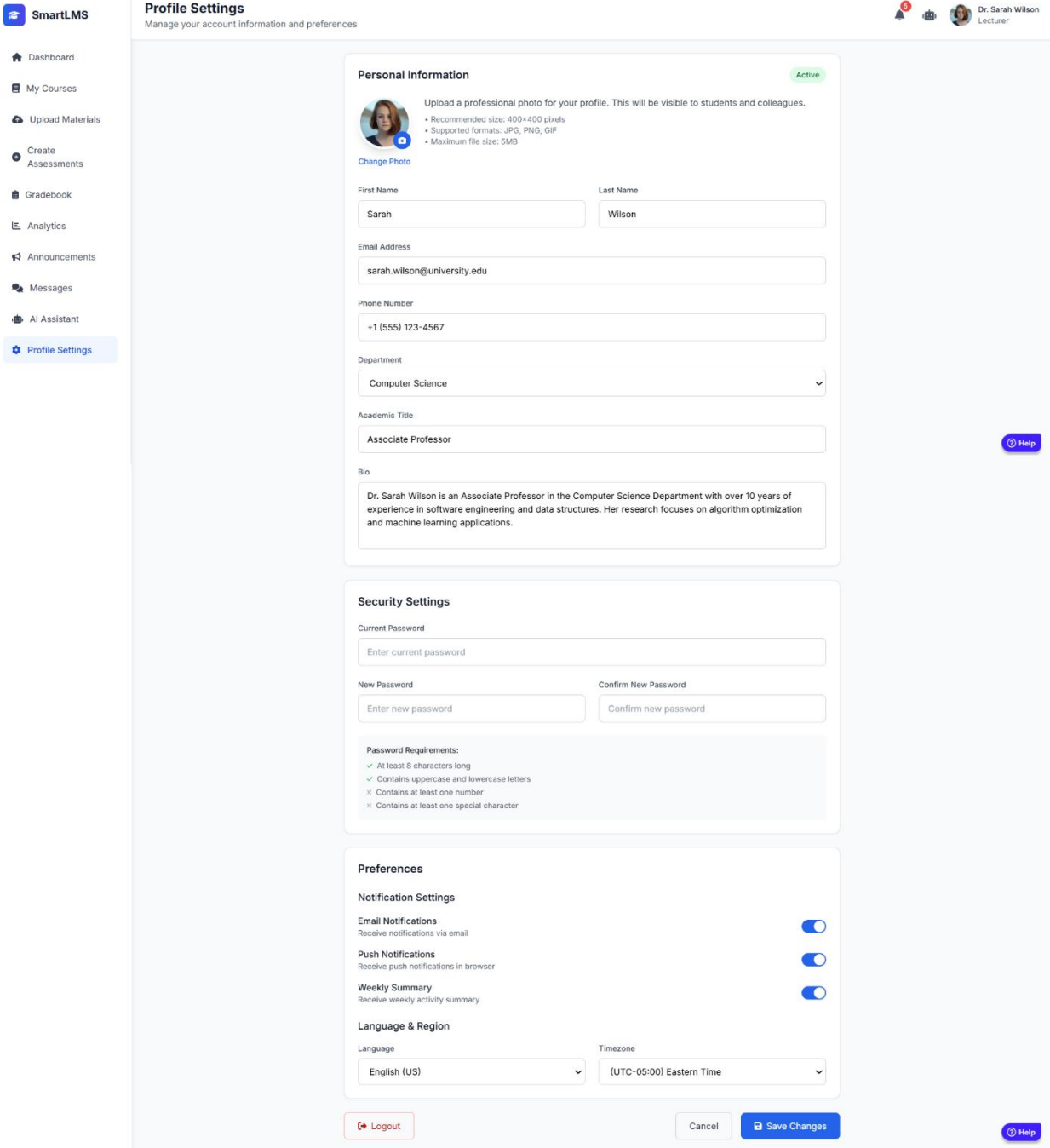


Fig 4.16 Profile page for lecturer to edit their personal information

Fig 4.16 illustrates the Lecturer Profile page where changes can be made to personal and security details.

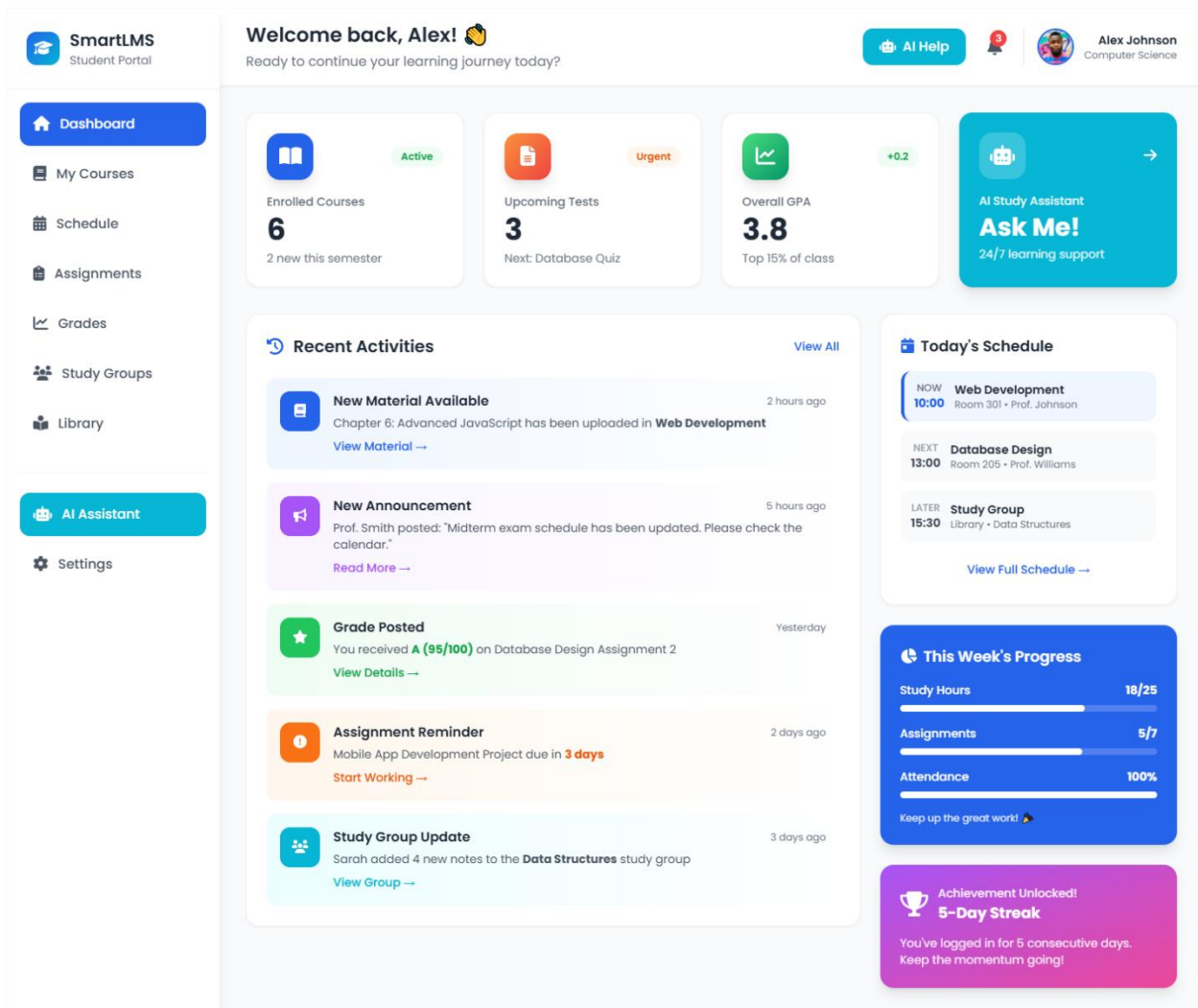


Fig 4.17 Student Homepage

Fig 4.17 illustrates the homepage layout of the student, showing a general overview of the student activities.

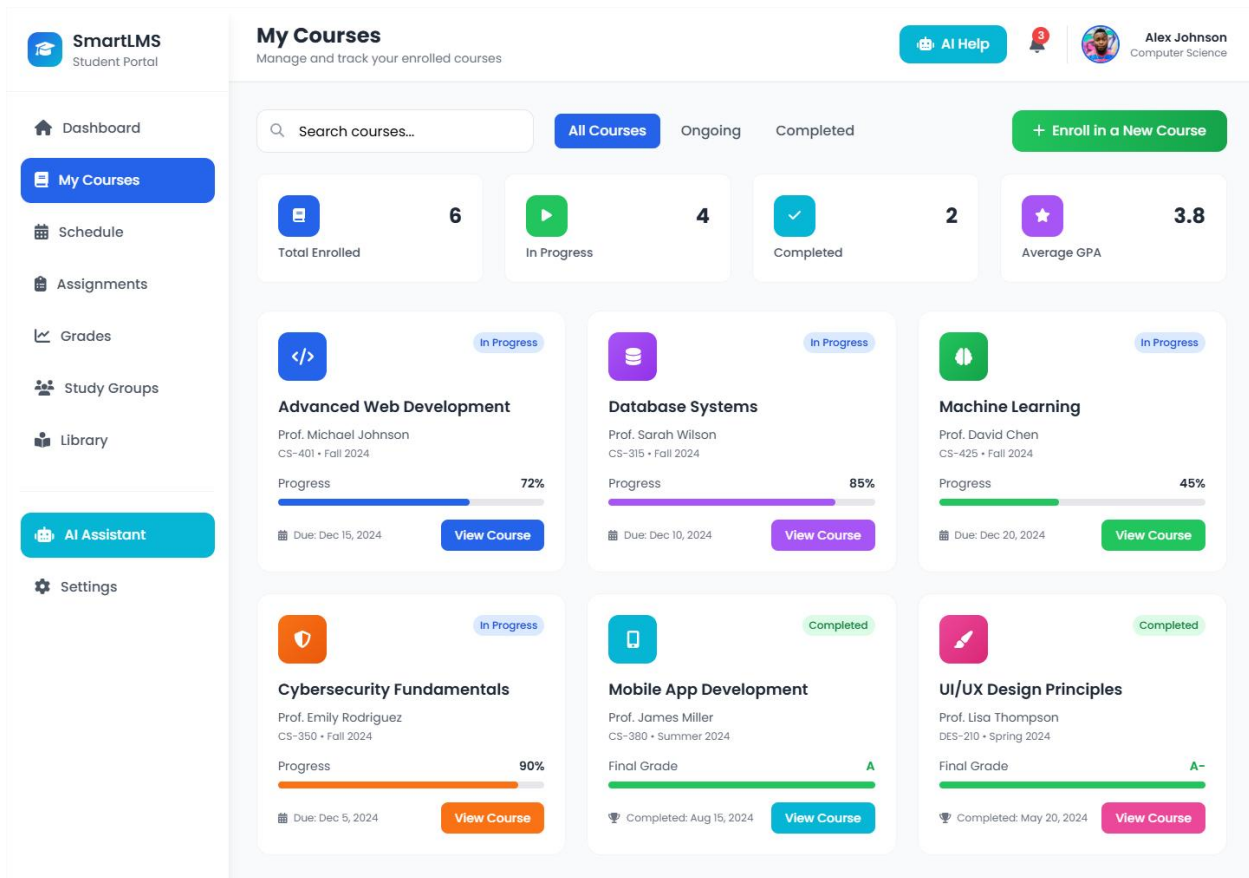


Fig 4.18 Student Enrolled Courses

Fig 4.18 shows the enrolled courses of the student. A student can enroll in different lecturer's course and all his enrolled courses will be displayed here on this page. The page shows the number of courses the student has enrolled in, the course cards and a button to enroll in a new course. To enroll in a new course, the student needs only the course code of the course (e.g CSC 423). It first checks if such course exists in the "course collection" in the database, if it does exist already, it throws an appropriate error, but if it doesn't, it proceeds to add the student's unique id to a collection called "enrolled students". And by that, if it turns out the student id is already in the "enrolled students" collection of the course, it throws an appropriate message that the student has already enrolled in this course.

The screenshot shows the SmartLMS interface for a course titled "Advanced Web Development". The page is divided into a left sidebar with navigation options like Dashboard, My Courses, Schedule, Assignments, Grades, Study Groups, and Library. The main content area features a course card with a progress bar showing 72% completion (8 of 12 modules). Below the card are tabs for Overview, Materials, Announcements, Assessments, and Lecturer Info. The Overview tab is active, displaying a course description, learning objectives (e.g., Master React.js, Build RESTful APIs), a weekly schedule (Lectures on Mondays/Wednesdays, Lab Sessions on Fridays), and a "Your Stats" section showing 94% attendance, 8/10 assignments, an A- current grade, and excellent participation. Upcoming deadlines include a Final Project due in 5 days (High priority) and Quiz 3 due in 2 weeks (Medium priority). AI suggestions for reviewing React Hooks and practicing with sample projects are also provided.

Fig 4.19 Profile page for lecturer to edit their personal information

Just like Fig 4.9, this Fig 4.19 illustrates the course details of each course. When the student clicks on any of the course card, this page is opened to show the overview of the course and everything else about the course.

The screenshot shows the 'Course Assessments' page for 'Advanced Web Development' in the SmartLMS Student Portal. The page is part of the 'My Courses' section, specifically for 'CS-401 • Prof. Michael Johnson'. The user is identified as 'Alex Johnson, Computer Science'.

The page features a navigation sidebar on the left with options: Dashboard, My Courses (selected), Schedule, Assignments, Grades, Study Groups, Library, AI Assistant, and Settings.

The main content area has tabs for Overview, Materials, Announcements, Assessments (selected), and Lecturer Info. A summary section displays: 12 Total Assessments, 4 Ongoing, 8 Completed, and an 87% Average Score.

Below the summary, there are filters for 'All', 'Ongoing', and 'Completed', along with a search bar and dropdowns for 'All Types' and 'Sort by Due Date'. The assessment list includes:

- Midterm Exam - Web Fundamentals** (DUE SOON): Due Nov 25, 2024 at 11:59 PM, 120 minutes, 40% of final grade. Actions: AI Study Help, Start Exam.
- Project Assignment - E-commerce Website** (IN PROGRESS): Due Dec 15, 2024 at 11:59 PM, Individual Project, 30% of final grade. Actions: AI Code Review, Continue.
- Quiz 4 - JavaScript Functions & Scope** (AVAILABLE): Due Nov 30, 2024 at 11:59 PM, 30 minutes, 5% of final grade. Actions: Practice Quiz, Start Quiz.
- Assignment 3 - CSS Grid & Flexbox** (COMPLETED): Submitted Nov 18, 2024, Grade: 92/100, 10% of final grade. Action: View Result (92%).
- Quiz 3 - HTML5 Semantic Elements** (COMPLETED): Submitted Nov 15, 2024, Grade: 88/100, 5% of final grade. Action: View Result (88%).
- Assignment 2 - Responsive Web Design** (COMPLETED): Submitted Nov 10, 2024, Grade: 95/100, 10% of final grade. Action: View Result (95%).
- Quiz 2 - CSS Fundamentals** (COMPLETED): Submitted Nov 8, 2024, Grade: 90/100, 5% of final grade. Action: View Result (90%).

Fig 4.20 Student Assessment Page

Fig 4.20 illustrates the Assessment Page of each student. It shows the assessment that has been created by the lecturer in each course, the assessment that has been attended to and the ones still under review. This page as shown above is still inside the “My Courses” page but in a another subsection.

The screenshot displays the SmartLMS interface for a course titled "Advanced Web Development" by Prof. Michael Johnson. The page is titled "Course Materials" and shows a list of 24 total materials. The materials are categorized by type and include options to view or download them. The materials listed are:

- Week 1 - Introduction to Modern Web Development** (PDF, 2.4 MB, 45 downloads)
- Lecture Slides - HTML5 & CSS3 Fundamentals** (PPTX, 5.8 MB, 52 downloads)
- Video Lecture - JavaScript ES6 Features** (Video, 45:23 mins, 38 views)
- Code Examples - React Component Patterns** (ZIP, 1.2 MB, 67 downloads)
- Reading Material - RESTful API Design Best Practices** (PDF, 3.1 MB, 41 downloads)
- Assignment Brief - Full Stack Web Application Project** (PDF, 890 KB, 58 downloads)
- Tutorial - Setting Up Node.js Development Environment** (Video, 28:45 mins, 44 views)
- Study Guide - Midterm Exam Preparation** (DOCX, 1.5 MB, 73 downloads)

Fig 4.21 Uploaded Materials Page

Fig 4.21 shows the materials that the lecturer has uploaded for a course. There are different materials that can be uploaded by the course lecturer which are, video, documents, images, slides, links. And all uploaded materials have an option to be downloaded to the local device. The page also shows the number of materials that has been uploaded for that course and some search and filter functionality to easily search and access a material.

Above are the proposed screens and pages for both the lecturer and the student. Some pages are very similar and only need to be tweaked a little to be used for either the student or the lecturer.

```
_id: ObjectId('6888db3768cf3f28117c4a28')
name: "Ryota Kise"
email: "kise_copyCat@gmail.com"
password: "$2b$10$TUh.xWwIYqfy54ULXFJF/.vzBR9t3QX0Fwe50.LpdFGV8mf9Ns200"
role: "student"
▶ notificationsSettings: Object
  createdAt: 2025-07-29T14:31:19.880+00:00
  updatedAt: 2025-07-30T17:28:45.030+00:00
  __v: 1
  passwordChangedAt: 2025-07-30T05:47:17.914+00:00
▶ enrolledCourses: Array (1)
```

Fig 4.22 Student Sample in the Database

Fig 4.22 above shows a student user in the database with its attributes, its unique id, name, email, password, role and enrolled courses.

```
_id: ObjectId('68619391ec8edfa75d902cfc')
courseTitle: "Operating Systems"
courseCode: "CSC411"
▶ lecturerIds: Array (2)
  createdAt: 2025-06-29T19:27:13.478+00:00
  updatedAt: 2025-08-03T03:53:24.015+00:00
  __v: 36
▶ classrooms: Array (1)
▶ materials: Array (2)
▶ summaries: Array (2)
```

Fig 4.23 Course Sample in the Database

Fig 4.23 shows how a course looks like in the database, showing the unique id, course code, course title, assessments, uploaded materials and announcements, etc.

```
▼ materials : Array (2)
  ▼ 0: Object
    title : "CSC 411 PQ Slip"
    type : "file"
    fileUrl : "https://res.cloudinary.com/djvinikc2/image/upload/v17514868..."
    description : "The 2023/2024 Past question paper for CSC 411"
    uploadedBy : ObjectId('6861878dbc4b322962febc2e')
    fileType : "image/jpeg"
    fileSize : 88981
    createdAt : 2025-07-02T20:09:34.780+00:00
    _id : ObjectId('686591fee9de4fc20d128092')
    updatedAt : 2025-07-02T20:50:52.560+00:00
```

Fig 4.24 Course Material Sample in the Database

Fig 4.24 shows the material uploaded for a course, file type and the file URL that directs to the file in the Cloudinary platform.

```
▼ enrolledStudents : Array (6)
  0: ObjectId('6888db3768cf3f28117c4a28')
  1: ObjectId('688a4bb9f65bf5ba4d19189a')
  2: ObjectId('688a4c57f65bf5ba4d19189d')
  3: ObjectId('688a4c9ff65bf5ba4d1918a0')
  4: ObjectId('688a4ce4f65bf5ba4d1918a3')
  5: ObjectId('688a4d28f65bf5ba4d1918a6')
```

Fig 4.25 Enrolled Students

Fig 4.25 shows the students that has enrolled in that particular course. With their unique id, they can be referenced in the “user” collection.

CHAPTER 5

SUMMARY, CONCLUSION AND RECOMMENDATION

5.1 BACKGROUND

This chapter summarizes the major findings, conclusions from the study, and practical recommendations based on the results. It also reflects on the accomplishments and limitations of the SmartLMS project and offers suggestions for future development and research.

The project aimed to design and implement a Smart Learning Management System (SmartLMS) that integrates Artificial Intelligence (AI) tools to improve learning experiences and support both lecturers and students. The sy

stem was evaluated to determine its effectiveness in tackling issues with traditional LMS platforms.

5.2 SUMMARY OF THE FINDINGS

The study aimed to develop a functional and intelligent Learning Management System that combines user-friendly design, AI-driven features, and effective academic management tools. The system was built to enhance digital learning experiences in institutions.

The most significant findings from this project include:

- i. **AI Integration Improved Learning Support:** The introduction of AI features, such as summarizing course materials and generating self-practice questions, significantly influenced student engagement and independent learning. These tools allowed students to study more effectively and helped lecturers gain insights into class performance.

- ii. **System Flexibility and Accessibility:** The SmartLMS system worked well on all mobile devices, including smartphones, tablets, and desktops. This made it easily accessible for all users.
- iii. **Enhanced Teaching:** Lecturers could create courses, upload materials, manage assessments, grade student work, and view performance analytics with ease.
- iv. **Data Security and Reliability:** Using secure authentication (JWT) and cloud-based storage (Cloudfire) ensured that user data and uploaded files were well protected.

5.3 CONCLUSION

The findings indicate that SmartLMS effectively addressed key limitations found in traditional Learning Management Systems. It created a more engaging, intelligent, and efficient digital learning environment for both lecturers and students.

This system shows that AI can positively transform education by enabling personalized learning and offering meaningful insights into academic performance. Importantly, the SmartLMS system ensures that AI features support and assist lecturers rather than replace them, with lecturers maintaining primary administrative rights.

The study also highlights the need for educational institutions, especially in developing contexts like Nigeria, to adopt more adaptable and AI-supported platforms that promote inclusivity, interactivity, and accessibility in learning. Overall, SmartLMS serves as an example of how technology can be effectively integrated into education to achieve better learning outcomes.

5.4 RECOMMENDATIONS

Based on the findings and outcomes of this study, the following recommendations are suggested:

- i. Enhancement of AI functionalities to include deeper student behavior analytics and personalized learning recommendations.
- ii. Integration of real-time chat and discussion features to foster better communication between lecturers and students.
- iii. The system should gradually be deployed to every department in the University of Benin for widespread use by everyone.
- iv. Proper orientation and training should be provided to both lecturers and students to ensure effective use of the platform.
- v. Further studies should explore the long-term impact of AI-driven LMS platforms on student performance and motivation.
- vi. Continuous feedback loops should be maintained with users to identify new needs and deploy new system updates.

5.5 SUMMARY

In summary, this study successfully designed and implemented SmartLMS, an AI-assisted learning management system that improves the teaching and learning experience in higher education for both lecturers and the students. The system demonstrated that integrating AI into educational technology can lead to more personalized, efficient, and engaging learning processes.

By combining intelligent tools, human control, and accessibility across platforms, SmartLMS has the potential to become a model for the next generation of academic learning systems. With further development and institutional adoption, it can contribute meaningfully to digital transformation in education, especially within universities in Nigeria and beyond.

REFERENCES

- Abdallah, A., El-Masri, M., & Badran, A. (2024). Artificial intelligence in higher education: Opportunities, challenges, and ethical implications. *Computers & Education*, 205, 105212. <https://doi.org/10.1016/j.compedu.2024.105212>
- Adamu, M., & Kamar, I. (2023). Evaluating the use of Moodle LMS in Nigerian universities during the COVID-19 pandemic. *International Journal of Educational Technology in Higher Education*, 20(1), 1–15.
- Adebayo, T., Omoregie, J., & Bello, S. (2023). Student satisfaction and e-learning adoption in Nigerian universities: A case study approach. *Journal of Educational Technology Research and Development*, 71(4), 999–1012.
- Akgun, S., & Greenhow, C. (2022). Artificial intelligence in education: Ethical implications and teacher perspectives. *Computers and Education: Artificial Intelligence*, 3(3), 100087. <https://doi.org/10.1016/j.caeai.2022.100087>
- Bada, S. O. (2015). Constructivism learning theory: A paradigm for teaching and learning. *Journal of Research & Method in Education*, 5(6), 66–70.
- Celik, I., Ozdemir, M., & Yilmaz, A. (2022). Enhancing teaching through digital learning platforms: The role of AI and analytics in education. *Education and Information Technologies*, 27(8), 10989–11007.
- Charitopoulos, K., Papadakis, S., & Tzavidas, D. (2020). Learning analytics in higher education: A systematic literature review. *International Journal of Educational Research*, 99, 101501.
- Chen, J., & Xie, H. (2022). Adaptive learning systems powered by AI algorithms: Challenges and applications. *Educational Technology & Society*, 25(3), 45–59.
- Chizobam, I. (2024). AI adoption and learning behavior among university students in Nigeria: A case of the University of Benin. *African Journal of Information Systems and Technology*, 13(2), 44–59.
- Costa, C., Alvelos, H., & Teixeira, L. (2017). The use of Moodle e-learning platform: A study in higher education. *International Journal of Educational Technology in Higher Education*, 14(1), 1–23.
- Dantas, L. M. V., & Cunha, C. J. C. D. A. (2020). The use of learning theories in education technology research: A bibliometric study. *British Journal of Educational Technology*, 51(6), 2000–2020.

Goksel, N., & Bozkurt, A. (2019). Artificial intelligence in education: Current insights and future perspectives. *International Journal of Research in Education and Science (IJRES)*, 5(2), 521–541.

Hwang, G. J., Xie, H., Wah, B. W., & Gašević, D. (2020). Vision, challenges, roles and research issues of artificial intelligence in education. *Computers and Education: Artificial Intelligence*, 1, 100001. <https://doi.org/10.1016/j.caeai.2020.100001>

Okafor, C., & Nwosu, A. (2022). Effectiveness of digital learning tools in Nigerian tertiary institutions: Opportunities and challenges. *African Journal of Educational Technology*, 9(1), 14–27.

Okonkwo, C. W., & Ade-Ibijola, A. (2021). Chatbots applications in education: A systematic review. *Computers and Education: Artificial Intelligence*, 2, 100011. <https://doi.org/10.1016/j.caeai.2021.100011>

Olowe, A. (2022). Challenges of e-learning implementation in Nigerian universities: A case study of the University of Benin. *Nigerian Journal of Information Systems and Education*, 8(2), 88–102.

Piaget, J. (1964). Development and learning. *Journal of Research in Science Teaching*, 2(3), 176–186.

Siemens, G. (2005). Connectivism: A learning theory for the digital age. *International Journal of Instructional Technology and Distance Learning*, 2(1), 3–10.

Sweller, J. (1994). Cognitive load theory, learning difficulty, and instructional design. *Learning and Instruction*, 4(4), 295–312. [https://doi.org/10.1016/0959-4752\(94\)90003-5](https://doi.org/10.1016/0959-4752(94)90003-5)

Wang, Y., Huang, R., & Lee, T. (2024). Artificial intelligence in education: The state of the art and future directions. *Educational Technology Research and Development*, 72(1), 57–81.

Zawacki-Richter, O., Marín, V. I., Bond, M., & Gouverneur, F. (2019). Systematic review of research on artificial intelligence applications in higher education: Where are the educators? *International Journal of Educational Technology in Higher Education*, 16(1), 1–27.