



Department of Electrical Engineering & Computing Sciences

Academic Year	2025-2026				
Semester	<input checked="" type="radio"/> Fall	<input type="radio"/> Spring	<input type="radio"/> Summer		
Course Code - Name	ISTE-500.600 Senior Development Project I				
Instructor	Huda Saadeh				
Exam	<input type="radio"/> Quiz/Test	<input type="radio"/> Assignment	<input type="radio"/> Mid	<input type="radio"/> Final	<input checked="" type="radio"/> Other
	Deliverable3 Report				
Start Date	Mon., Nov. 10, 2025				
Duration	Due Friday. Dec 12, 2025				

- Write and submit deliverable3 report based on report template guidelines and your mentors, instructor's guidance
- The report should include all sections mentioned in syllabus for deliverable3

Student ID	Student Name
396001477	Jeeda Kotob
396004854	Ryma Ait Tayeb
771008583	Ibraheem Mustafa

CLOs:

1. Apply requirements elicitation methods in an extra-classroom environment.
2. Architect an effective, user-centric solution.
3. Apply contemporary software development practices.
4. Create and document a proper project plan.

If the assessment maps to a single CLO, state the CLO below:

CLO	Section	Max Score	Students Average Score
CLO4	All report	20	

# **CourseGenie+: Centralized Academic Workflow Automation**

by

Jeeda Kotob, 396001477 , CIT  
Ryma Ait Tayeb, 396004854, CIT  
Ibraheem Mustafa, 771008583, CIT

A Capstone Project Submitted to RIT-Dubai

Fall 2025

Mentors

Dr. Ali Assi  
Mr. Qusai Hasan

## ABSTRACT

Learning management systems are an essential component of higher education institutions. Existing platforms often prioritize student services and tend to underrepresent requirements from instructors and administrators. We present CourseGenie+, a web-based application with customized dashboards that streamlines faculty workflows by automating manual and time-consuming tasks. The system integrates syllabus and assessment management, grading statistics, course peer reviewing, exam scheduling, room distributions, and reporting tools for course and program evaluation. CourseGenie+ also allows administrators to track syllabus submissions, collect Course Assessment Reports, configure course sections and manage faculty workflow. Integration of CourseGenie+ establishes it as a centralized academic workflow automation tool that bridges the gap between teaching, administration, and accreditation, enabling faculty to focus more on teaching while ensuring administrators meet compliance and quality standards.

### **Keywords**

Learning Management Systems, Workflow Automation, Academic Administration, Dashboards, Course Assessment Reports, Course Learning Outcomes, Program Learning Outcomes.

## TABLE OF CONTENTS

List of Figure.....	vi
List of Tables .....	vii
I. Introduction .....	1
II. Problem Statement .....	2
III. Norms and Values.....	3
A. Work and Quality.....	3
B. Collaboration and Participation.....	3
C. Accountability .....	3
IV. Literature Review .....	4
A. Introduction.....	4
B. Features and Comparisons .....	4
C. Gaps and Limitations in LMSs.....	5
D. Visualization and Analytics in LMSs .....	6
E. Assessment and Accreditation Needs.....	8
F. Peer Review in Academic Systems .....	9
G. Conclusion .....	11
V. Benchmarking .....	12
VI. Requirements Elicitation Methods .....	13
A. Brainstorming.....	13
B. Observations.....	13
C. Interviews .....	13
1) Interviews with Professors.....	13
2) Interview with Institutional Effectiveness Manager.....	13
VII. Project Scope.....	14
A. Included Scope.....	14
B. Excluded Scope .....	14
VIII. Development Methodology and Process.....	15
A. Team Roles and Responsibilities .....	15
B. Task Tracking and Progress Management .....	15
IX. Stakeholders.....	16
A. Primary.....	16

1) Professors.....	16
2) Administrators .....	16
B. External .....	16
1) Accreditation Bodies .....	16
X. System Requirements and Documentation .....	17
A. Functional and Non-Functional Requirements .....	17
1. Functional Requirements .....	17
2. Non-Functional Requirements.....	18
B. User Stories .....	19
C. Use Cases .....	43
1. User Login & Dashboard Access.....	43
2. System-Generated CAR Preview & Submission.....	45
XI. System Design and Architecture .....	47
A. System Overview .....	47
B. System Use Case Diagram .....	47
C. Database Design.....	48
D. User Interface Design.....	49
E. Behavioral Design .....	52
1. Sequence Diagrams.....	52
2. Activity Diagrams.....	55
XII. Configuration Requirements and Technologies .....	57
XIII. Project Planning .....	58
A. Work Breakdown Structure .....	58
B. Gantt Chart .....	59
1. Progress Reflection .....	59
C. RACI Matrix .....	60
XIV. Risk Assessment .....	60
XV. Resources and Budget Planning.....	63
A. Bill of Materials (BoM) .....	63
B. Budget Summary.....	65
XVI. Testing Plan .....	65
XVII. Ethics and Impact on Society .....	68
XVIII. Stakeholder Feedback, Reflection and Lessons Learned.....	69

XIX. Conclusion .....	70
References.....	71

## LIST OF FIGURES

Figure 1: Trello Board for Task Tracking.....	15
Figure 2: System Use Case Diagram .....	47
Figure 3: Entity-Relationship Diagram.....	48
Figure 4: Landing Page .....	49
Figure 5: User Login Interface.....	49
Figure 6: Professor Dashboard Overview .....	50
Figure 7: Course Overview Page (Professor View).....	50
Figure 8: Role Selection Screen for Multi-Role Users .....	51
Figure 9: Administrator Dashboard Overview .....	51
Figure 10: Sequence Diagram - User Login & Dashboard Access .....	52
Figure 11: Sequence Diagram - Syllabus Submission Tracking .....	53
Figure 12: Sequence Diagram - CAR Preview & Submission .....	53
Figure 13: Sequence Diagram - Exam Room Assignment .....	54
Figure 14: Activity Diagram - User Login & Dashboard Selection .....	55
Figure 15: Activity Diagram - Syllabus Submission Tracking.....	55
Figure 16: Activity Diagram - CAR Preview & Submission .....	56
Figure 17: Activity Diagram - Exam Room Assignment .....	56
Figure 18: Work Breakdown Structure.....	58
Figure 19: Gantt Chart .....	59
Figure 20: RACI Matrix.....	60

## LIST OF TABLES

Table 1: Project Benchmark.....	12
Table 2: Project User Stories .....	42
Table 3: UC01 - User Login & Dashboard Selection .....	44
Table 4: UC02 - System-Generated CAR Review & Submission.....	46
Table 5: 5x5 Risk Assessment Matrix .....	61
Table 6: Risk Level Interpretation Legend .....	61
Table 7: Project Risk Register & Mitigation Strategies.....	63
Table 8: BoM .....	64
Table 9: System Test Plan.....	68

## I. Introduction

Learning Management Systems (LMSs) have become essential tools in higher education systems, providing digital support for teaching, learning, and administration [1]. At RIT, the LMS used is MyCourses by Brightspace. MyCourses supports students through their learning journey, by providing features such as assignment submission, content delivery, grade viewing and attendance tracking. These features are also provided for instructors, however, they are considered limited compared to the required teaching workflows.

To fill in these gaps, faculty often tend to rely on external solutions such as spreadsheets, email, online review forms and text reports. According to professors and academic stakeholders interviewed during the requirements elicitation phase, these methods require extra manual work and are time consuming, error prone, and difficult to manage for accreditation purposes.

CourseGenie, a previous capstone project, demonstrated the potential of extending the LMS functionality with a narrowed focus on professor centered tools as assessment management, syllabus generation and grading statistics. However, the system still lacks many advanced features and several unresolved issues.

CourseGenie+ introduces dedicated dashboards for both instructors and administrators. These dashboards will centralize the academic workflow and automate key academic tasks including, but not limited to, Course Assessment Report (CAR) file generation, course calendar generation, exam management, etc. By creating role-specific dashboards, CourseGenie+ aims to increase efficiency, reduce manual work, prevent errors and provide an enhanced visibility to course and program level performance.

## II. Problem Statement

Within higher education institutions, there is growing pressure on faculty members and educational administrators to carry out an ever expanding number of repetitive, time consuming academic management activities. These include drafting and collecting syllabi, compiling CAR files, mapping Course Learning Outcomes (CLOs) to Program Learning Outcomes (PLOs), and managing examination logistics such as scheduling and room allocation. Additional requirements, such as peer review documentation, further add to the administrative responsibilities.

Currently, most of these activities are completed manually through emails, spreadsheets, and limited LMS tools. This fragmented approach results in inefficiencies, delays, and a lack of centralized oversight. For administrators, the absence of automated systems to track progress or alert relevant personnel often leads to missed deadlines, delayed course reviews, and difficulties in ensuring compliance with accreditation standards.

The core problem lies in the lack of centralized, automated tools to support professor and administrator workflows in existing LMS platforms, particularly MyCourses. Without such tools, inefficiencies persist, compliance becomes difficult, and academic staff are unable to fully focus on teaching and institutional quality improvement.

### III. Norms and Values

The team has established clear norms and values to be adapted by the team to ensure consistent quality, accountability, and collaboration throughout the project.

#### *A. Work and Quality*

All deliverables are expected to be completed, clean, and professional, reflecting both qualitative and quantitative effort. The team places emphasis on meaningful progress and constructive feedback, ensuring that all work contributes to the overall success of the project.

#### *B. Collaboration and Participation*

Tasks are assigned by skills, but all members are responsible for understanding all aspects of the project. This guarantees that during assessments and presentations, every member can answer questions and contribute fully. Open discussions in weekly meetings and ongoing communication through group chats provide space for all members to share ideas, raise issues, and propose solutions.

#### *C. Accountability*

Punctual attendance at team meetings is required, with any exceptions communicated in advance. Each member is responsible for meeting deadlines and delivering high quality work. Team members are expected to take ownership of tasks, demonstrate initiative, and commit fully to the project, going beyond minimum requirements where necessary.

## IV. Literature Review

### *A. Introduction*

Learning Management Systems play a critical role in the higher education learning systems affecting all aspects of teaching, learning and administration. Several systems have been implemented and have reached a universal level. However, the literature shows that these systems vary significantly in terms of usability, reporting capabilities and support for different stakeholders. This review explores five major themes that have been identified across recent studies: core features and comparisons, common gaps and limitations in current LMSs, system visualization and analytics, assessment and accreditation requirements, and peer review in academic systems. These themes highlight the strengths and limitations of existing learning management systems which will aid in the process of delivering CourseGenie+.

### *B. Features and Comparisons*

One of the most recurrent conclusions in the literature is that learning management systems offer a “core toolkit” of features. Sánchez et al. [1] conducted a comparative study of 45 LMSs based on six categories: interoperability, accessibility, productivity tools, communication tools, learning tools, and security. They found that almost all systems included common functions such as assignment submissions, content repositories, announcements, and gradebooks. These are considered essential for course delivery and serve as the foundation of current LMSs. However, the authors also showed that systems differed significantly in adaptability and usability. For example, Moodle and Paradiso were rated as the strongest overall, as they offered greater adaptability and smoother integration of tools across categories. Brightspace, on the other hand, scored only average, providing the core toolkit but with less flexibility and limited support for

broader institutional workflows. The authors concluded that effectiveness depends not only on features themselves, but also on how well tools align with institutional needs and instructor workflows.

Another common theme is the tradeoff between administrative strength and usability. Yamani, Alharthi, and Smirani [2] compared Brightspace and Blackboard with surveys of over 500 students. They found that Brightspace was preferred for its clean, user-friendly interface, while Blackboard was rated higher for its reporting and administrative capabilities. This variation reflected differences in design priorities: Blackboard emphasized administration and refined reporting, while Brightspace prioritized student accessibility and ease of use. At RIT Dubai, where MyCourses is based on Brightspace, this trade-off means students benefit from simplicity, but instructors and administrators face limitations in reporting and course management. CourseGenie+ aims to fill this gap by maintaining usability while introducing professor and administrator-oriented tools such as CAR file generation, and exam scheduling.

These studies collectively show that while contemporary LMSs meet a baseline of shared features, they tend to prioritize students over faculty. This leaves instructors without sufficient tools for accreditation related tasks or advanced analytics. For CourseGenie+, this evidence highlights the value of extending MyCourses by building on the core toolkit with enhanced visualization, reporting, and accreditation tools.

### *C. Gaps and Limitations in LMSs*

While LMSs are effective at delivering core teaching tools, multiple studies highlight persistent gaps in how they serve instructors and administrators. These gaps arise not only from missing features but also from the limited adoption and usability of tools that already exist.

Simon et al. [3], in a mixed methods study with 41 instructors and 41 students, found that Moodle was primarily used as a repository for slides and announcements. Instructors avoided its collaboration features, preferring email or WhatsApp, while students reported built-in tools were confusing and time consuming. The authors concluded that tool utilization depends more on alignment with teaching practices than on availability.

Li, Jung, and Wise [4] studied instructor engagement with learning analytics dashboards and found two patterns: some teachers attempted to adapt their teaching, while others only monitored performance. Yet, in both cases, dashboards were difficult to interpret, and instructors lacked institutional support to use the data effectively. The authors argued that dashboards often fail when they present complex data without actionable context.

Together, these studies show that LMSs often provide advanced features that are underutilized or ineffective in practice. Instructors rely on external tools for communication and spend extra time manually interpreting abstract data. For CourseGenie+, these insights reinforce the need to avoid duplicating replaceable features (e.g., chat forums) and instead focus on automating essential workflows such as CLO–PLO mapping, syllabus tracking, and CAR reporting. Its contextualized grade visualization and statistics aim to reduce manual effort while directly supporting instructor decision making.

#### *D. Visualization and Analytics in LMSs*

The use of visualization and analytics in Learning Management Systems has grown to be essential for improving efficiency and vital decision making. Hernández-de-Menéndez et al. [5] explained that learning analytics allows educators to gather, process, and interpret large amounts of data to better evaluate performance, boost student engagement, and refine teaching strategies.

Their research highlights that well-designed visualization dashboards can turn complex datasets into clear, actionable insights. These tools help instructors and administrators recognize trends, monitor progress, and make more informed decisions. This idea directly connects to the goals of CourseGenie+, which plans to include interactive dashboards that display grading trends, assessment completion, and course review updates for both professors and administrators.

Ngulube and Ncube [6] further emphasize that analytics should not only measure academic performance but also enhance the overall user experience (UX) of LMS platforms. Their review found that features such as progress charts, heatmaps, and interactive dashboards greatly improve engagement and usability, especially for instructors. When analytics are integrated in intuitive and concise ways, they argue that they help reduce mental effort and ease users in finding helpful insights quickly. In the same way, CourseGenie+ uses visual analytics within its professor and admin dashboards to simplify everyday tasks, such as monitoring syllabus submissions, tracking CAR file progress, and analyzing assessments.

Kui et al. [7] conducted a comprehensive survey of visual analytics techniques that are used in online education. They concluded that visual analytics bridge untapped educational data with actionable decision making. The authors describe different visualization models, like temporal, comparative, and hierarchical, that enable educators to monitor student progress, recognize struggling learners, and analyze learning outcomes. They state that visual analytics encapsulates the intersection of untapped educational data and actionable data driven decisions. Following the same logic, CourseGenie+ incorporates similar analytics to not only generate student performance visualization but also attend to institutional needs such as CLO–PLO mapping and monitoring course performance trends. These are the studies reinforcing the value

that analytics and visualization bring toward enhancing the quality of teaching and streamlining the administrative functions of an institution. Still, the majority of legacy LMS systems continue to prioritize the student-facing analytics while the faculty workflow components, like assessment documentation and peer review tracking, remain underappreciated. Learning analytics in CourseGenie+ expands on this paradigm by offering comprehensive tools to document transparent, goal directed, and collaborative efforts to bridge faculty and administrators.

#### *E. Assessment and Accreditation Needs*

Assessment and accreditation are central to quality assurance in higher education, ensuring that academic programs meet established educational standards. Vlachopoulos et al. [8] conducted a systematic review on “genuine assessment” and emphasized the importance of evaluation methods that reflect real world outcomes and generate evidence-based data for accreditation. Their evaluation covers over a decade of research and shows the need and value of digital assessment systems in providing accurate, evidence-based data to accrediting authorities. This is consistent with CourseGenie+, which aims to improve consistency and transparency in the reports by automating CAR files and linking CLOs with PLOs. The same review pointed out that traditional assessment and reporting methods are often slow, inconsistent, and error-prone, creating major challenges during accreditation audits. They call for integrated digital systems that can centralize evidence, automate report generation, and simplify faculty participation within outcome-based education (OBE) frameworks. CourseGenie+ answers this need by offering a unified platform that streamlines CAR & Syllabus files management and submissions, CLO–PLO mappings, grade statistics and more. Automating these processes reduces administrative effort while maintaining the precision and accountability necessary for accreditation. The review further emphasizes that authentic assessment should go beyond checking boxes for compliance, and it should encourage

continuous improvement and reflective teaching. This necessitates systems that capture data over extended periods and associate assessment data with changes in the curriculum [8].

To summarize, while the literature demonstrates how technology has transformed assessment and accreditation, it also points to the absence of integrated solutions that consolidate all these disparate elements. CourseGenie+ addresses these issues by having a comprehensive and data-empowered solution that leverages valuable assessment, continuous improvement, and accreditation readiness to ensure the establishment is always prepared.

#### *F. Peer Review in Academic Systems*

A study done by Jenkins et al. [9] identified how a structured faculty peer review process can enhance teaching quality and institutional culture. The study was conducted at Johns Hopkins School of Nursing, where they used the Design for Six Sigma (DMADV) framework: Define, Measure, Analyze, Design, Verify. The paper focused on the first three stages by conducting methods like surveys (Define), literature benchmarking (Measure), and taskforce discussions (Analyze), which helped to identify best practices in peer review of teaching [9]. These principles can guide the implementation of a digital peer review feature, where structured workflows, automated report generation, and survey integration ensure the process is systematic and transparent.

In the define phase, the Faculty Peer Review of Teaching Taskforce was created, composed of 11 members across different academic ranks and disciplines. Through earlier literature research, the group identified four key themes: (1) peer observation is useful but needs a supportive environment, (2) success requires clear goals, consistent tools, and trained reviewers, (3) post-observation discussions are essential, and (4) faculty ownership increases adoption. Benchmarking

across eight peer institutions revealed substantial variations, with some having formal models and others using peer review informally. This highlights the lack of a standardized model across higher education. CourseGenie+ could use these findings by embedding standardized peer review templates and structured discussion fields into its professor dashboard, improving consistency and collaboration.

To understand faculty opinions, the researchers conducted a Qualtrics survey on “qualities of good teaching” and “peer review procedures.” Out of 68 full-time faculty, 27 responded ( $\approx 40\%$ ), strongly supporting ideas like active learning, student–faculty interaction, prompt feedback, and inclusiveness in teaching. These findings aligned with Chickering and Gamson’s (1987) seven principles of good practice. While most faculty agreed with the teaching-quality indicators, weaker areas were related to online teaching and time management, showing that digital or hybrid instruction needs more institutional support. In CourseGenie+, similar surveys can be built directly into the system to collect feedback, with results visualized and linked to peer review reports to create a continuous improvement loop.

Most faculty also supported a formative and recurring peer review approach rather than a one-time evaluation. About 81% saw reviews as developmental, 93% preferred reviewers with several years of teaching experience, and all agreed that new faculty should receive a review within two years. Many suggested repeating reviews every 3–5 years and coordinating them through the Office of Teaching and Learning. This shows that faculty prefer a system focused on mentorship, reflection, and collaboration instead of ranking. The CourseGenie+ platform can apply this by automatically scheduling review cycles, sending alerts, and keeping data confidential, ensuring the process is efficient and consistent.

The main limitation of the study is that it was conducted at one institution with only full-time faculty, excluding adjunct and part time instructors whose roles often differ. Despite this, the research provides a strong foundation for building peer review systems in higher education. For CourseGenie+, this means developing flexible tools that support all types of faculties and can be scaled across diverse academic contexts.

#### *G. Conclusion*

In conclusion, from the literature work, we noticed a recurring pattern that most LMS platforms provide a common set of features, such as gradebooks, submissions, and content repositories. However, they tend to fail to meet instructor and administrative needs. Comparative studies emphasized usability trade offs, with systems like Brightspace favoring student experience but limiting advanced reporting and administrative workflows. Other studies showed that even when analytics and collaboration tools are present, they are often underutilized due to complexity or lack of alignment with educational needs. Together, these results highlight major weaknesses in the current LMS architecture and highlight the need for solutions that go beyond features that directly affect students to facilitate faculty collaboration, accreditation, and administrative management.

## V. Benchmarking

This section compares existing LMS platforms to identify gaps in faculty centered workflows. Table 1 highlights the key features and limitations relevant to the design of CourseGenie+.

Platform / System	Core Features	Limitations Identified	Relevance to CourseGenie+
Brightspace (MyCourses)	Official LMS used at RIT; supports assignment submission, grading, announcements, and attendance tracking	Lacks comprehensive faculty tools such as CAR file generation, syllabus tracking, and peer review management	Serves as integration reference; highlights the need for professor and admin centered automation
Blackboard	Provides advanced administrative and reporting tools	Complex interface and less user friendly for instructors	Demonstrates the trade off between functionality and usability
Moodle	Open source LMS with customizable modules and plugin support	Limited data visualization and less intuitive analytics dashboards	Serves as a model for extensibility but indicates need for improved reporting and visualization
CourseGenie (Phase 1)	Previous capstone project with core syllabus and assessment management	Missing automation features for CAR files, peer review workflows, and exam management	Establishes the baseline system; informs the iterative enhancements in <i>CourseGenie+</i>

Table 1: Project Benchmark

Table 1 shows that existing LMS platforms either focus primarily on student facing features or provide administrative capabilities with limited usability. These gaps motivated the design of CourseGenie+, which emphasizes automated faculty and administrative workflows.

## VI. Requirements Elicitation Methods

### *A. Brainstorming*

The team conducted brainstorming sessions in the initial weekly meetings alongside the project mentors. These discussions helped gather project requirements from the mentors. In addition, the brainstorming sessions also allowed the team members to collaborate with mentors in the process of identifying new features for the project.

### *B. Observations*

The team closely inspected the existing CourseGenie Phase 1 system to identify implemented features, gaps, and usability issues. Observing how some features were implemented revealed some bugs and limitations based on the requirements. In addition, the team reviewed the university's current LMS platform, MyCourses, from a faculty perspective to better understand the limitations and benchmark potential features for the project.

### *C. Interviews*

*1) Interviews with Professors:* Weekly meetings with project mentors Dr. Ali Assi and Mr. Qusai Hassan served as informal interviews for the requirement gathering stage. In these sessions, the mentors introduced the limitations of the current system being used, MyCourses, and the academic processes outside the lecture halls. They also provided suggestions on what enhancements and functionalities they expect in their roles as Professors at RIT Dubai.

*2) Interview with Institutional Effectiveness Manager:* To capture the administrative perspective, the team interviewed Ms. Dezzil Castelino, Institutional Effectiveness Manager at RIT Dubai. As a potential administrative user role in the proposed system, she highlighted the need for features such as an admin dashboard, syllabus submission tracking, CAR files collection and

initiation of peer review processes. This ensured that the system requirements addressed not only professor needs but also administrative oversight.

## VII. Project Scope

### *A. Included Scope*

The scope of CourseGenie+ focuses on extending existing LMS functionalities with features specifically targeted toward professors and administrators. For professors, the scope includes tools such as assessment management, syllabus generation, grade visualization and statistics, CLO management, CAR file generation, CLO–PLO mapping, course review reports, and exam related management (room allocation, schedules, notifications). Additionally, CourseGenie+ will provide a customizable course calendar with drag and drop functionality and professor profile management.

For administrators, the scope includes a dedicated dashboard to configure courses and sections, monitor syllabus submission, collect CAR files, and initiate peer review processes. These features will ensure that CourseGenie+ serves as a centralized and streamlined platform for academic management while complementing existing student-oriented systems like MyCourses.

### *B. Excluded Scope*

To avoid duplication of functionality already well served by MyCourses, CourseGenie+ will not include student dashboard features such as assignment submissions, gradebooks, attendance and course content view.

## VIII. Development Methodology and Process

This project is managed using the Agile Scrum methodology, allowing iterative development and continuous feedback from project mentors and stakeholders.

### A. Team Roles and Responsibilities

Role	Assigned Members
Product Owner	Dr. Ali Assi, Mr. Qusai Hasan
Scrum Master	Jeeda Kotob
Development	Jeeda Kotob (Full Stack), Ryma Ait Tayeb (Backend), Ibraheem Mustafa (Frontend)
Stakeholders	Professors, Administrators (IE Office)

### B. Task Tracking and Progress Management

Trello is used as the primary tool for tracking development tasks and monitoring progress throughout the project. Tasks are organized into three main lists: Product Backlog, In Progress, and Product Increment.

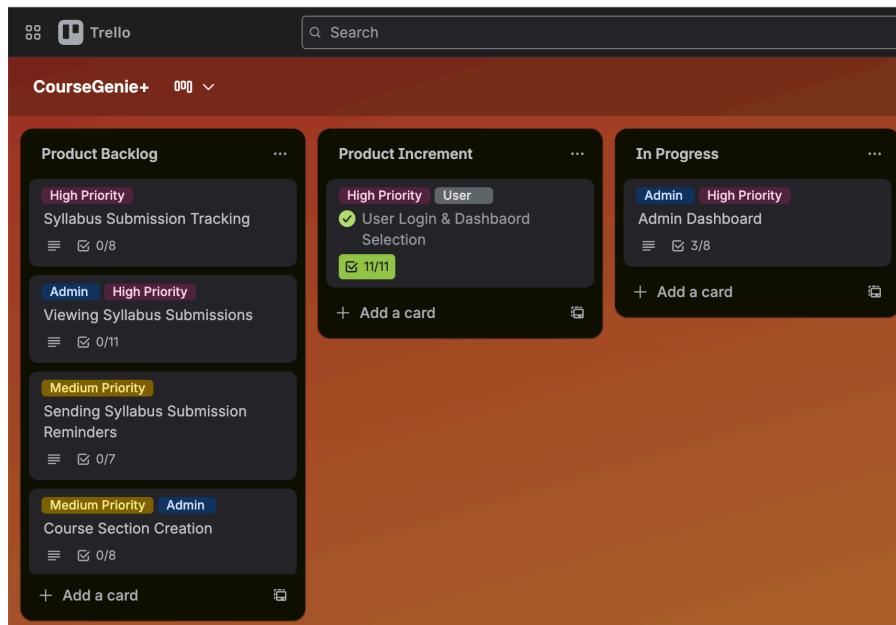


Figure 1: Trello Board for Task Tracking ([Trello Board](#))

## IX. Stakeholders

Our project involves multiple stakeholders with different priorities and interests. These stakeholders can be classified into primary, secondary and external groups based on their level of involvement and interaction with the system.

### *A. Primary*

*1) Professors:* They are the main users of the system. Professors use the dashboards to create and manage assessments, generate syllabi, visualize grades, and produce CAR files.

*2) Administrators:* Represented by the Institutional Effectiveness Manager and related staff, administrators use the system to configure courses, track syllabus submission, collect CAR files, and manage the peer review process.

### *B. External*

*1) Accreditation Bodies:* Organizations such as the KHDA, set the standards for curriculum alignment and reporting. While they do not interact with the system directly, their requirements influence features such as CLO–PLO mapping, CAR reports, and syllabus submission tracking.

## X. System Requirements and Documentation

This section outlines the functional & non-functional requirements, user stories, and use cases that define how CourseGenie+ operates and benefits system users.

### *A. Functional and Non-Functional Requirements*

The following system requirements define the core features CourseGenie+ must provide and the performance and security constraints it must satisfy.

#### *1. Functional Requirements:*

- The system must allow professors to generate and manage assessments with weights and descriptions.
- The system must allow professors to edit syllabi content and regenerate the file.
- The system must display student grades with visualization and summary statistics.
- The system must generate CAR files including CLO–PLO mapping and performance metrics.
- The system must provide professor profiles that can be edited (office hours, office location, email).
- The system must support cross-course collaboration by displaying professors teaching the same course and enabling communication.
- The system must support exam management by distributing students into rooms, generating reports, and sending notifications with exam date, time, and room details.
- The system must generate a course calendar automatically from assessments and allow drag and drop updates.

- The system must allow professors to create course review reports for remarks, suggestions, and peer review
- The system must allow admins to configure which courses professors can view.
- The system must track syllabus submission progress and display completion status.
- The system must provide a centralized collection of CAR files.
- The system must allow admins to initiate the peer review process, assign reviewers, and set deadlines.

## *2. Non-Functional Requirements:*

- Given that multiple professors access the system simultaneously, the system must handle the load without crashing or significant performance degradation.
- Given that a professor or admin performs an action, when the action is being executed, then the system must respond within 2 seconds.
- Given that professors and admins may access the system across different devices, when the system is used on desktops, tablets, or mobiles, it must remain accessible and usable.
- Given that academic data is sensitive, when users log in or perform actions, the system must ensure session management and security to protect confidentiality.

## B. User Stories

The user stories capture the system from each user's perspective. Table 2 summarizes the prioritized user stories and the acceptance criteria.

Title	User Story	Acceptance Criteria	Priority	Estimate
1. User Login & Dashboard Selection	<p>As a user who may have one or multiple roles (Admin and/or Professor), I want to log in and when applicable, choose my role so that I am taken directly to the corresponding dashboard.</p>	<p>Given the user has Professor role only, when the user logs in successfully, then the user is navigated directly to the Professor Dashboard.</p> <p>Given the user has Admin role only, when the user logs in successfully, then the user is navigated directly to the Admin Dashboard.</p> <p>Given the user has both Admin and Professor</p>	High	5

		<p>roles, when the user logs in successfully, then the system displays a role-selection popup with options “Continue as Admin” and “Continue as Professor”</p> <p>Given the role-selection popup is displayed, when the user selects “Continue as Professor”, then the user is navigated directly to the Professor Dashboard.</p> <p>Given the role-selection popup is displayed, when the user selects “Continue as Admin”, then the user is navigated directly to the Admin Dashboard.</p>		
--	--	--	--	--

2. Admin Dashboard	<p>As an admin, I want to have an administrative dashboard, so that I can manage courses and sections efficiently.</p>	<p>Given the admin is on the admin dashboard page, when the page loads, then the dashboard should display courses as clickable cards.</p> <p>Given the dashboard displays course cards, when the admin clicks on a course card, then the admin should be navigated to the course page showing its section cards.</p>	High	5
3. Syllabus Submission Tracking	<p>As an admin, I want to view syllabi submission progress bars for professors, so that I can easily track uncompleted syllabus submissions.</p>	<p>Given an admin on the admin dashboard, when the admin selects the syllabus tab, then the system should redirect the admin to the Syllabus Page.</p>	High	5

		<p>Given an admin is on the Syllabus Page, when the page loads, then the admin should see progress bars displaying professors' syllabus submission progress, organized by department.</p>		
4. Viewing Syllabus Submissions	<p>As an admin, I want to view syllabus submissions in a designated tab so that I don't have to manually collect submissions.</p>	<p>Given the admin is on the syllabus page, when the admin clicks on a professor's name, then the system should display all syllabi submitted by that professor.</p> <p>Given the admin is viewing a professor's syllabus submissions, when the admin selects</p>	High	8

	<p>a syllabus, then the system should open a preview of the selected syllabus for review.</p> <p>Given the admin is viewing a professor's syllabus submissions, when the admin clicks the download button, then the system should download the file in PDF format.</p> <p>Given a professor has no syllabus submissions, when the admin views that professor's record then the system should indicate that no submissions are available.</p>	
--	--	--

		Given the admin is viewing the list of syllabus submissions for a professor, when a submission's timestamp is later than the designated deadline, then the system should display an 'Overdue' indicator next to that syllabus entry.		
5. Sending Syllabus Submission Reminders	As an admin, I want the system to automatically send reminders to professors who have not submitted their syllabi so that I don't have to manually follow up.	Given a professor has not submitted the syllabus file, when the due date is 2 days away, then the system should automatically send an email reminder that includes the course section and due date.	Medium	3

6. Course Section Creation	<p>As an admin, I want to configure new course sections from the system so that I can manage semester schedules efficiently.</p>	<p>Given an admin is viewing a course page, when the page loads, then an “Add Section” button is displayed.</p> <p>Given an admin is on the course page, when the admin clicks “Add Section”, then a create section form popup is opened.</p>	Medium	3
7. CAR File Submission Tracking	<p>As an admin, I want to view progress bars for professors, so that I can easily track uncompleted CAR file submissions.</p>	<p>Given an admin on the admin dashboard, when the admin selects the CAR Files tab, then the system should redirect the admin to the CAR files page.</p> <p>Given an admin is on the CAR files page, when the page loads,</p>	High	5

		then the admin should see progress bars displaying professors' CAR files submission progress, organized by department.		
8. Viewing CAR File Submissions	As an admin, I want to view CAR file submissions in a designated tab so that I don't have to manually collect submissions.	<p>Given the admin is on the CAR File page, when the admin clicks on a professor's name, then the system should display all CAR files submitted by that professor.</p> <p>Given the admin is viewing a professor's CAR file submissions, when the admin selects a file, then the system should open a preview of the selected CAR for review.</p>	High	8

		<p>Given the admin is viewing a professor's CAR submissions, when the admin clicks the download button, then the system should download the file in PDF format.</p> <p>Given a professor has no CAR file submissions, when the admin views that professor's record then the system should indicate that no submissions are available.</p> <p>Given the admin is viewing the list of CAR File submissions for a professor,</p>		
--	--	---	--	--

		when a submission's timestamp is later than the designated deadline, then the system should display an 'Overdue' indicator next to that CAR entry.		
9. Sending CAR Submission Reminders	As an admin, I want the system to automatically send reminders to professors with uncompleted CAR submissions, so that I don't have to manually follow up with them.	Given a professor has not submitted the CAR file, when the due date is 2 days away, then the system should automatically send an email reminder that includes the course section and due date.	Medium	3
10. Peer Review Reports Submission Tracking	As an admin, I want to view progress bars for professors, so that I can easily track uncompleted peer	Given an admin on the admin dashboard, when the admin selects the Peer Review tab, then the system should	Medium	5

	<p>review report submissions.</p>	<p>redirect the admin to the peer review page.</p> <p>Given an admin is on the peer review page, when the page loads, then the admin should see progress bars displaying professors' peer review submission progress, organized by department.</p>		
11. Exam Room Allocation	<p>As an admin, I want to assign available rooms to each course's exam in the system, so that I can eliminate the manual process of sending room lists, collecting responses and resolving conflicts.</p>	<p>Given the admin is on the Exam Room page, when the admin selects one or more rooms for a course and clicks Save, then the selected rooms are assigned to that course in the system.</p> <p>Given the room allocations have been</p>	High	5

		saved, when professors view the exam room assignment page, then they only see the rooms allocated to their course.		
12. System-Generated CAR Preview & Submission	As a professor, I want the system to automatically generate the CAR file for my course section so that I can reduce manual effort, minimize errors and streamline the reporting process.	Given the professor is on a course section page, when they open the CAR Report tab, then the system generates and displays a CAR preview with available course data.  Given the CAR preview is displayed and required data is missing, when the professor views the preview, then the system indicates missing information and	High	8

		<p>disables the Submit button.</p> <p>Given the CAR preview is displayed, when the professor selects Download, then the system generates and downloads the CAR as PDF.</p> <p>Given the CAR preview is displayed and required data is complete, when the professor selects Submit, then the system records the CAR submission and displays a confirmation message.</p>		
13. Peer Review Evaluation (Reviewer)	As a reviewer, I want to complete my peer review evaluation	Given the reviewer is logged in, when they open the peer review	Medium	5

	<p>within the system through a guided workflow so that I can provide structured and unbiased feedback without coordination delays.</p>	<p>page, then the system displays a “Pending Reviews” section listing professors they must review.</p> <p>Given a pending review is listed, when the reviewer selects “Start Review”, then the system displays the evaluation form with required rating and comment fields.</p> <p>Given the reviewer completes all required fields, when the reviewer submits the evaluation, then the system saves the evaluation and displays a confirmation message.</p>		
--	--	--	--	--

		Given required fields are missing, when the reviewer attempts to submit, then the system prevents submission and shows a warning.		
14. Peer Review Evaluation (Reviewee)	As a reviewee, I want to access the peer review feedback I received and submit my reflection separately so that I can provide unbiased responses without time constraints.	Given the reviewer has submitted feedback, when the reviewee navigates to their peer review page, then the system displays a “Received Feedback” section containing the submitted evaluation.  Given the feedback is visible, when the reviewee selects “Add Reflection” the system displays a text field for entering reflection.	Medium	5

		Given the reviewee enters a reflection, when they submit the reflection, then the system saves the reflection and displays a confirmation message.		
15. User Profile	<p>As a user, I want to view and update my personal information such as office hours and contact details on my dashboard so that I can keep my profile accurate and up to date.</p>	<p>Given a user is on any page, when the user clicks the avatar in the top right, then a menu options with 'Edit Profile' option.</p> <p>Given the menu is open, when the user clicks Edit Profile, then the profile page opens displaying current user info.</p>	Low	3

		Given the profile page is open, when the user edits the fields and clicks Save, then the changes are saved and shown on the profile.		
16. Course Calendar Generation	As a professor, I want all course assessments to be automatically added to my course calendar and be able to modify them when needed so that I can maintain an organized and up to date schedule.	Given a professor is on a course overview page, when they toggle the sidebar menu, they should see a “Course Calendar” option listed.  Given a professor is on a course overview page, when they click on the “Course Calendar” option from the sidebar, then they should be redirected to the calendar page for that specific course section.	High	8

	<p>Given a professor is on the calendar page, when the page loads, then a weekly course calendar is displayed with labeled week cards containing assessment cards assigned to that week.</p> <p>Given the course calendar page is displayed, when there are unassigned assessments, then they appear in an Unscheduled Assessments section.</p> <p>Given the calendar and assessment cards are displayed, when the professor drags and drops an assessment to</p>		
--	---	--	--

		<p>any week, then the card appears under that week.</p> <p>Given the professor has moved assessment cards, when they click Save, then the updated placements are saved.</p>		
17. Exam Room Assignment	<p>As a professor, I want to assign my section's students to the available exam rooms in the system so that the allocation is completed correctly without needing manual student distribution or coordination with other instructors.</p>	<p>Given the rooms are assigned to the course, when the professor opens Exam Room Assignment, then the available rooms and their capacities are displayed.</p> <p>Given the professor selects one or more rooms, when they confirm the selection then the system automatically distributes</p>	High	8

		<p>the enrolled students across the selected rooms based on capacity.</p> <p>Given the professor has confirmed the selection, when they click save, then the room assignment is saved and visible to other instructors.</p>		
18. Exam Notifications	As a professor, I want students to automatically receive email notifications with the assigned exam room, date, and time so that I can ensure timely and accurate communication of exam details.	Given the professor has completed the exam room assignment for their section, when the professor clicks save, then the system automatically generates and sends an email notification to all enrolled students containing exam date,	Medium	3

		time and their assigned exam room.		
19. Exam Room Allocations Report	As a professor, I want to access and print a report showing students assigned exam rooms so that I can easily verify allocations and assist with on-site coordination during exams.	Given the exam room assignment has been completed, when the professor clicks view report, then a report should be displayed showing each student and their assigned room.  Given the report is displayed, when the professor clicks download, then the report should be exported in a printable format.	Medium	5
20. Cross-Course Collaboration	As a professor, I want to view professors teaching the same courses and have direct links for	Given a professor is on a course overview page, when the page loads, then a card is displayed listing other professors	Low	3

	<p>individual and group emails so that we can collaborate more efficiently and maintain alignment in course delivery.</p>	<p>teaching the same course.</p> <p>Given the collaboration card is displayed, when a professor clicks an individual name or the group email option, then an email window opens addressed to that professor or group.</p>		
21. Grade Coloring	<p>As a professor, I want grades in the gradebook to be automatically color coded highlighting top, average and bottom performers per assessment so that I can quickly identify performance trends without manual review.</p>	<p>Given a professor is on a course overview page, when they toggle the sidebar menu, they should see a “Grading” option listed.</p> <p>Given a professor is on a course overview page, when they click on the “Grading” option from the sidebar, then they</p>	High	5

	<p>should be redirected to the grade book page for that specific course section.</p> <p>Given a professor is on the grading page, when the page loads, then a gradebook table is displayed showing students as rows and assessments as columns.</p> <p>Given the gradebook is displayed, when assessment scores appear, then the two highest scores in that column are highlighted in green, the two scores closest to average are highlighted in yellow, and the two lowest</p>		
--	--	--	--

		scores in that column are highlighted in red.		
22. Grade Statistics	As a professor, I want to view grade statistics such as class averages, grade distributions, and pass rates so that I can analyze overall class performance and identify areas for improvement.	<p>Given a professor is on a course overview page, when they toggle the sidebar menu, they should see a “Grade Statistics” option listed.</p> <p>Given a professor is on a course overview page, when they click on the “Statistics” option from the sidebar, then the dashboard should display overall class performance statistics such as average score, highest score, lowest score, pass/fail rate, grade distribution, etc.</p>	Medium	5

Table 2: Project User Stories

### C. Use Cases

The use cases specify the flow of interactions between users and the system for two of the system's key workflows.

#### 1. User Login & Dashboard Access

Name	User Login & Dashboard Access
Identifier	UC01
Description	The system authenticates a user and navigates them to the appropriate dashboard based on their assigned role(s). If the user has both Admin and Professor roles, the system prompts the user to select which dashboard to access.
Goal	Ensure authenticated users are directed to the correct dashboard based on their role(s).
Actors	User (Admin, Professor or both), Authentication System
Preconditions	<ol style="list-style-type: none"><li>1. The user account exists</li><li>2. The user's role assignments are defined in the system</li></ol>
Assumptions	The login page shows username, password, a log in button and a reset password link.
Frequency	Multiple times per day during active semesters.
Main Flow	<ol style="list-style-type: none"><li>1. User enters login credentials and submits login request.</li><li>2. The system authenticates the credentials.</li><li>3. The system retrieves the role(s) assigned to the user.</li></ol>

	<p>4. The system navigates the user to the appropriate role-specific dashboard.</p> <p>5. The use case ends when the appropriate dashboard is displayed.</p>
Alternate Flows	<p>AF 2.1: Invalid Credentials (At step 2)</p> <ul style="list-style-type: none"> <li>○ 2a. The system displays an error message (“Invalid username or password”).</li> <li>○ 2b. The user may try again or click Reset Password.</li> </ul> <p>AF 3.1: User Has Multiple Roles (At step 3)</p> <ul style="list-style-type: none"> <li>○ 3a. The system displays a role-selection popup with options (“Continue as Admin” / “Continue as Professor”).</li> <li>○ 3b. The user selects the desired role for the current session.</li> <li>○ 3c. The system proceeds to step 4, redirecting to the chosen role’s dashboard.</li> </ul>
Post Conditions	<ul style="list-style-type: none"> <li>● Success: The user is authenticated and redirected to the role-specific dashboard.</li> <li>● Failure: The user remains on Login page with an error message shown.</li> </ul>

*Table 3: UC01 - User Login & Dashboard Selection*

## 2. System-Generated CAR Preview & Submission

Name	System-Generated CAR Preview & Submission
Identifier	UC02
Description	When a professor selects one of their course sections and opens the CAR tab, the system automatically generates and displays a preview of the CAR. The professor may download the CAR or submit it if all required data is complete.
Goal	Provide professors with a generated CAR that reduces manual reporting effort and supports review, download, and submission.
Actors	Professor (Primary) , Database (Supporting System)
Preconditions	<ol style="list-style-type: none"> <li>1. The professor is logged in.</li> <li>2. The professor has at least one assigned course section.</li> <li>3. CAR generation data sources (grades, CLO mapping, assessment records, etc.) exist in the database.</li> </ol>
Frequency	Preview can be accessed multiple times during semester. Submission is only done once per semester.
Main Flow	<ol style="list-style-type: none"> <li>1. The professor logs in and is navigated to the Professor Dashboard.</li> <li>2. The professor selects one of the course sections displayed on the dashboard.</li> <li>3. The professor opens the CAR tab.</li> <li>4. The system generates and displays the CAR preview for the selected section.</li> <li>5. The use case ends when the CAR preview is visible.</li> </ol>

Alternate Flows	<p>AF 4.1: Incomplete Required Data (At step 4)</p> <ul style="list-style-type: none"> <li>○ 4a. The system detects that required CAR data is missing.</li> <li>○ 4b. The system displays a notice indicating missing information.</li> <li>○ 4c. The Submit action is disabled until the missing information is completed.</li> <li>○ 4d. The flow returns the professor reviewing the preview.</li> </ul> <p>AF 4.2: Download CAR (At step 4)</p> <ul style="list-style-type: none"> <li>○ 4a. The professor selects Download</li> <li>○ 4b. The system generates the CAR as a PDF.</li> <li>○ 4c. The system initiates the download.</li> <li>○ 4d. The flow ends, the professor remains on the CAR preview.</li> </ul> <p>AF 4.3: Submit CAR (At step 4)</p> <ul style="list-style-type: none"> <li>○ 4a. The professor selects Submit</li> <li>○ 4b. The system records the CAR submission.</li> <li>○ 4c. The system displays a confirmation message.</li> <li>○ 4d. The flow ends, the professor remains on the CAR preview.</li> </ul>
Post Conditions	<ul style="list-style-type: none"> <li>● Preview only (no action taken): The CAR preview is displayed.</li> <li>● Preview with successful download: The CAR preview is displayed, and a CAR PDF successfully downloaded.</li> <li>● Submission Recorded: The CAR was submitted with confirmation.</li> <li>● Preview with submission unavailable and warning message.</li> </ul>

Table 4: UC02 - System-Generated CAR Review & Submission

## XI. System Design and Architecture

### A. System Overview

CourseGenie+ is a client-server web application consisting of an Angular frontend, a Spring Boot backend, and a relational database. The backend follows a layered MVC pattern, separating controllers, services, and data access components. Communication between the frontend and backend is handled through RESTful APIs.

### B. System Use Case Diagram

The system use case diagram provides a high level overview of the CourseGenie+ system. It identifies the primary system actors and their interactions with the key use cases of the system highlighting use case dependencies and role-based access workflows within a shared system.

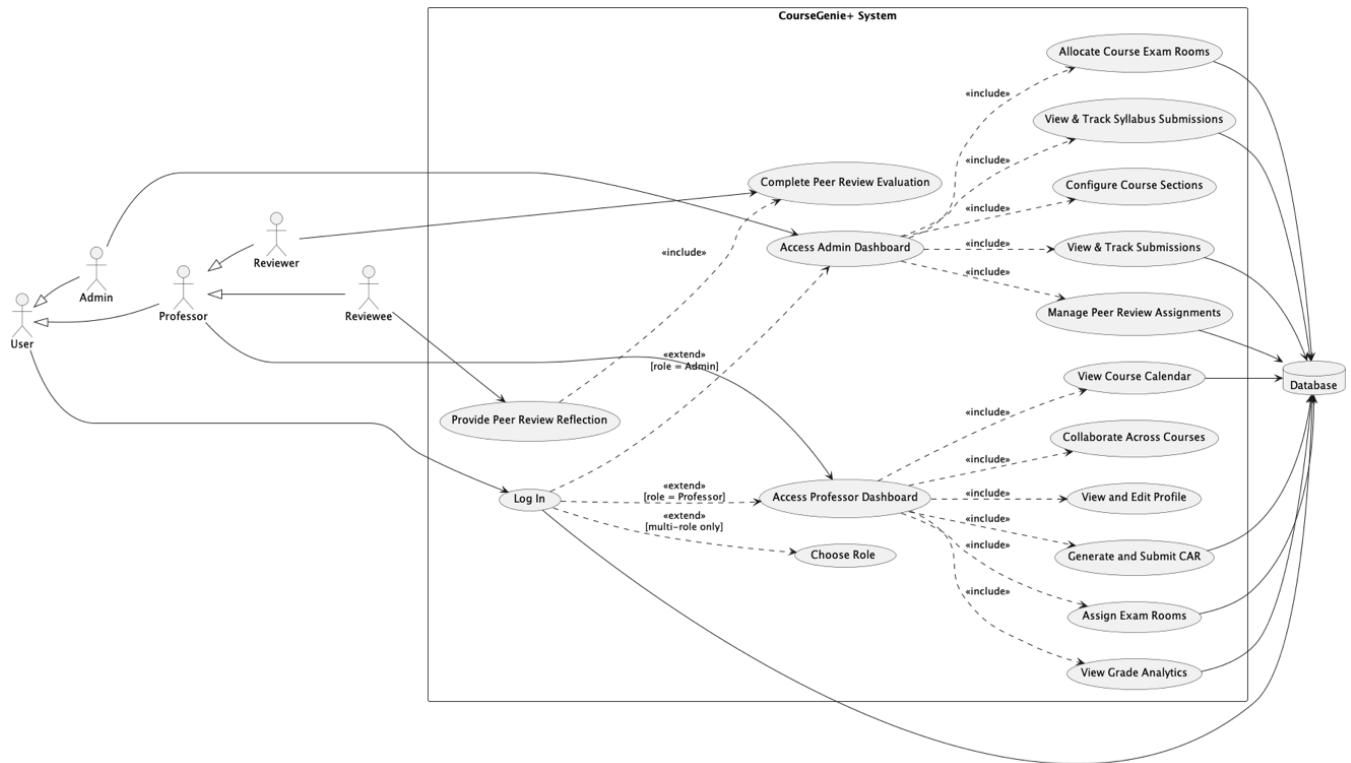


Figure 2: System Use Case Diagram

### C. Database Design

The database structure of CourseGenie+ is represented using an entity–relationship diagram (ERD) that illustrates the core entities, attributes, and relationships supporting the system.

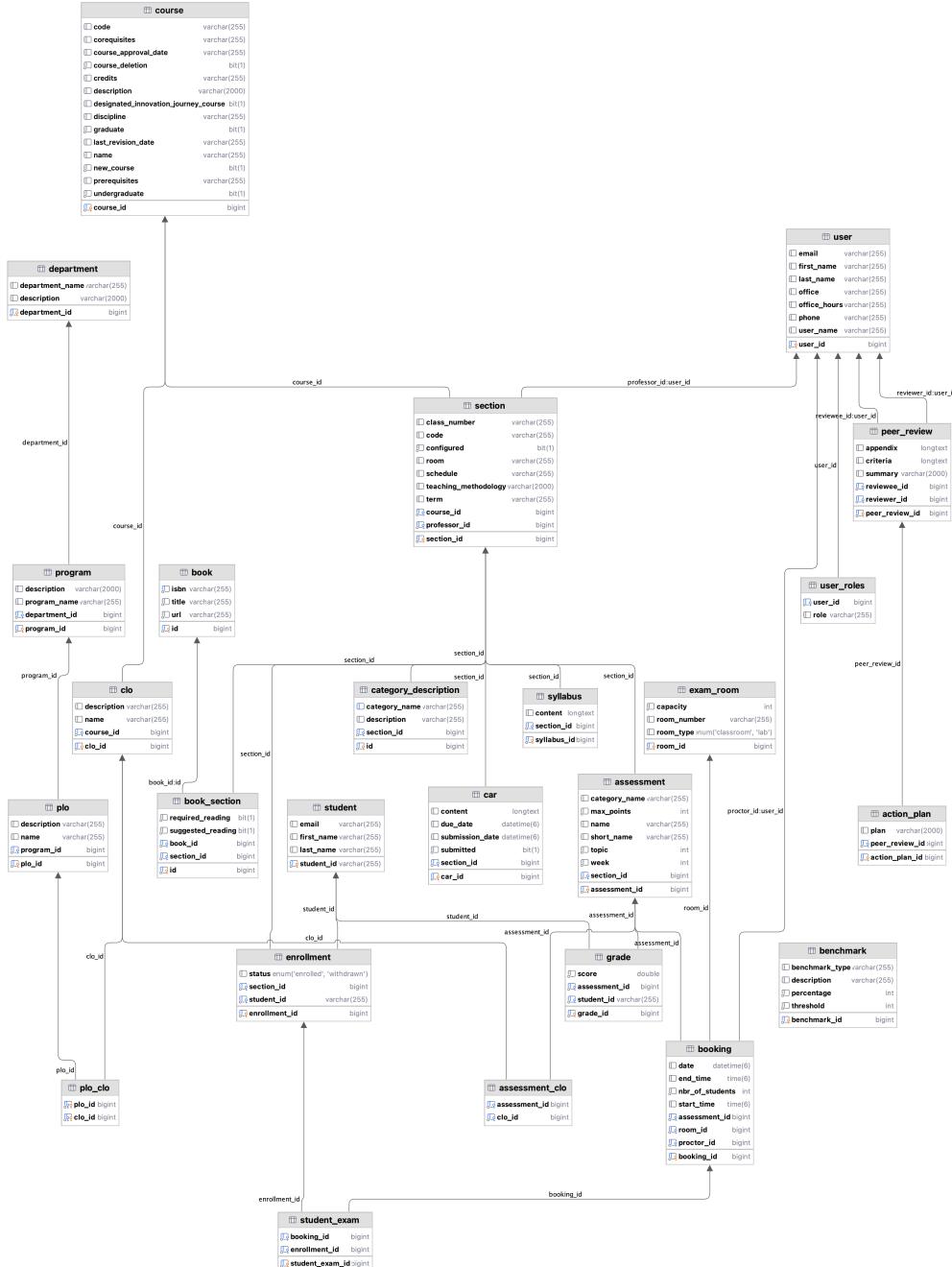


Figure 3: Entity-Relationship Diagram

## D. User Interface Design

This section presents representative screenshots of the CourseGenie+ user interface. The designs demonstrate role-based navigation, dashboard layouts, and primary user entry points implemented during the current development phase. The interfaces emphasize clarity, usability, and consistency for both professors and administrators.

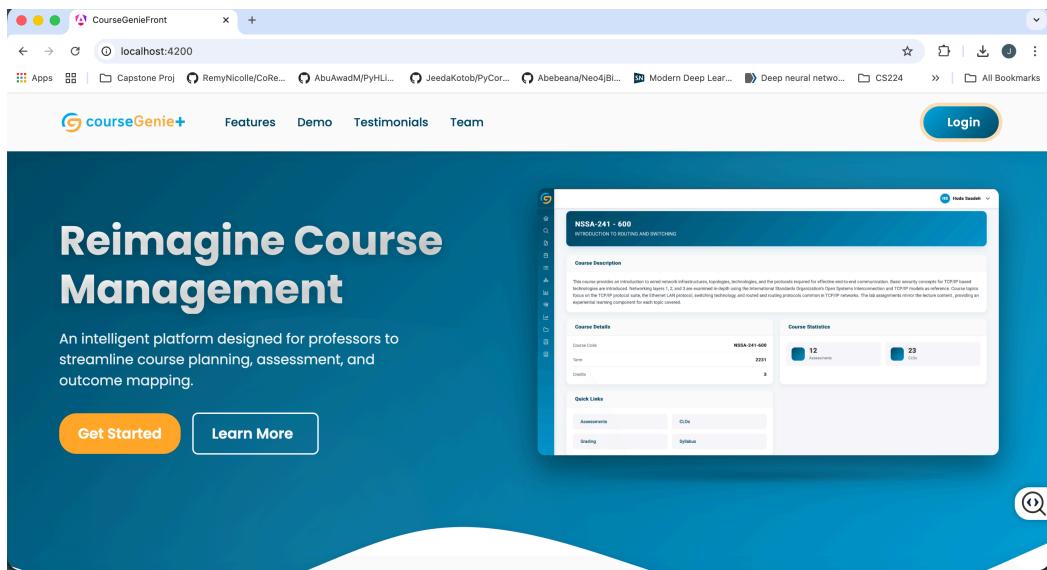


Figure 4: Landing Page

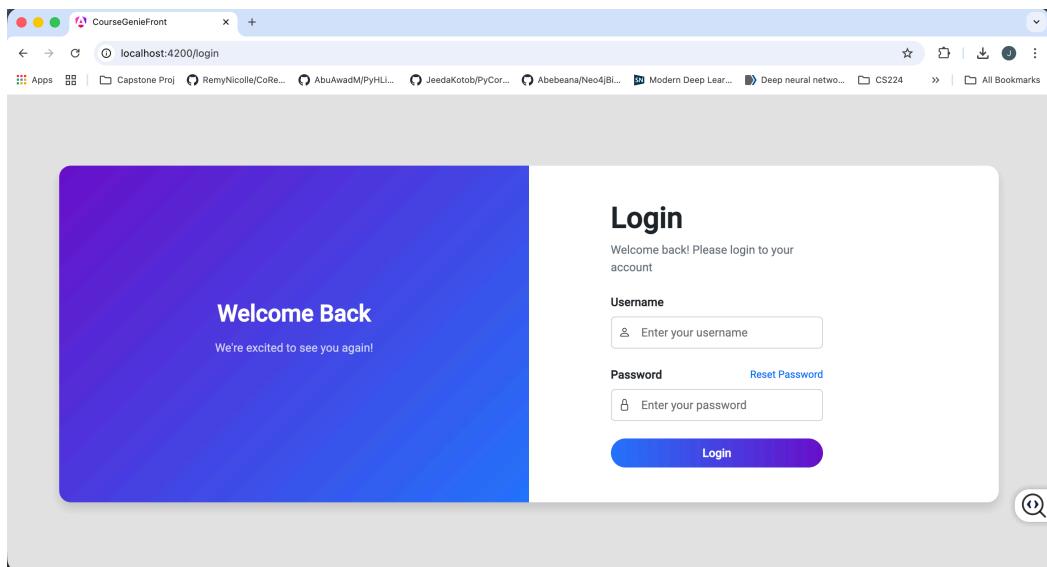


Figure 5: User Login Interface

The screenshot shows a browser window titled "CourseGenieFront" with the URL "localhost:4200/professor". The page has a header with the text "to manage Assessments, CLOs and much more ...". Below the header, there is a section titled "Your courses" with a dropdown menu set to "All Semesters". Under "Fall 2025", there are four course cards:

- Software Development and Problem Solving I** (4 Credits) - GCIS 123 - 602, Fall 2025, Section 602. Buttons: "View Course" and a magnifying glass icon.
- Introduction to Database and Data Modeling** (3 Credits) - ISTE 230 - 600, Fall 2025, Section 600. Buttons: "View Course" and a magnifying glass icon.
- Introduction to Database and Data Modeling** (3 Credits) - ISTE 230 - 602, Fall 2025, Section 602. Buttons: "View Course" and a magnifying glass icon.
- Introduction to Software Engineering** (3 Credits) - SWEN 261 - 600, Fall 2025, Section 600. Buttons: "View Course" and a magnifying glass icon.

Figure 6: Professor Dashboard Overview

The screenshot shows a browser window titled "CourseGenieFront" with the URL "localhost:4200/overview/GCIS%20123/602". The page has a sidebar with icons for home, search, file, etc. At the top right, there is a user profile for "Ali Assi". The main content area shows course details for "GCIS 123 - 602" (Software Development and Problem Solving I). It includes a "Course Description" section with a paragraph about algorithmic thinking and Python programming, and "Course Details" and "Course Statistics" sections. The "Course Details" section shows "Course Code: GCIS 123-602" and "Term: Fall 2025". The "Course Statistics" section shows "1 Assessments" and "1 Mapped CLOs".

Figure 7: Course Overview Page (Professor View)

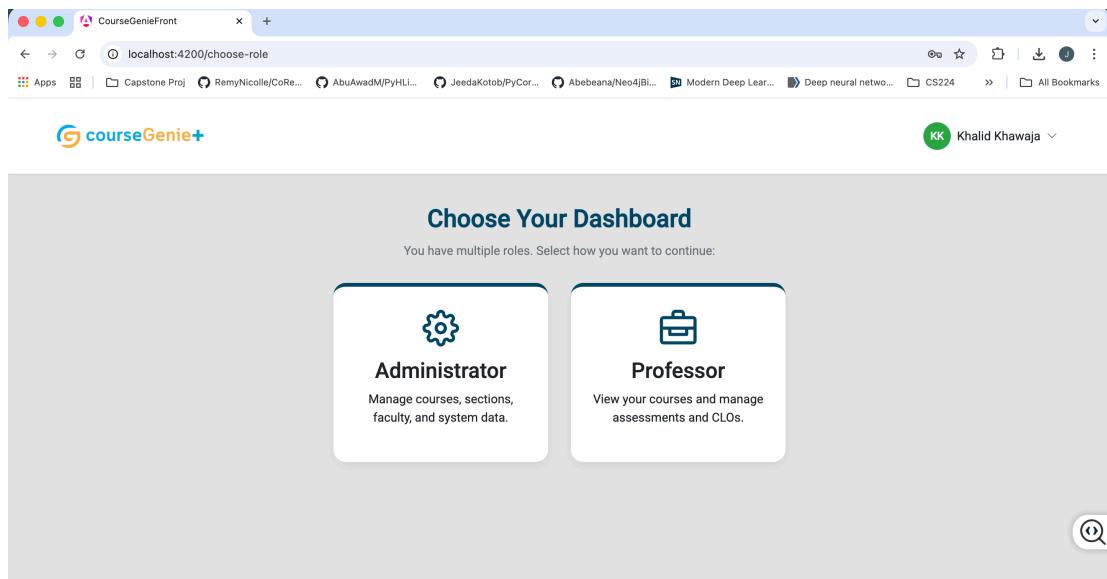


Figure 8: Role Selection Screen for Multi-Role Users

A screenshot of the "Administrator" dashboard, titled "All Courses". The page displays a grid of eight course cards, each containing course details, credits, and a "View Course" button. The courses listed are: Introduction to Database and Data Modeling (3 Credits, ISTE 230), Software Development and Problem Solving I (4 Credits, GCIS 123), Computer System Concepts (3 Credits, NSSA 102), Software Development and Problem Solving II (4 Credits, GCIS 124), Web &amp; Mobile I (3 Credits, ISTE 140), Introduction to Routing &amp; Switching (3 Credits, NSSA 241), Task Automation with Interpreted Languages (3 Credits, NSSA 220), and Web &amp; Mobile II (3 Credits, ISTE 240). Each card also includes a small icon and a "View Course" button.

Figure 9: Administrator Dashboard Overview

## E. Behavioral Design

*1. Sequence Diagrams:* The sequence diagrams illustrate the interaction flow between system components for selected core user stories, highlighting the order of operations between the user interface, backend services, and database.

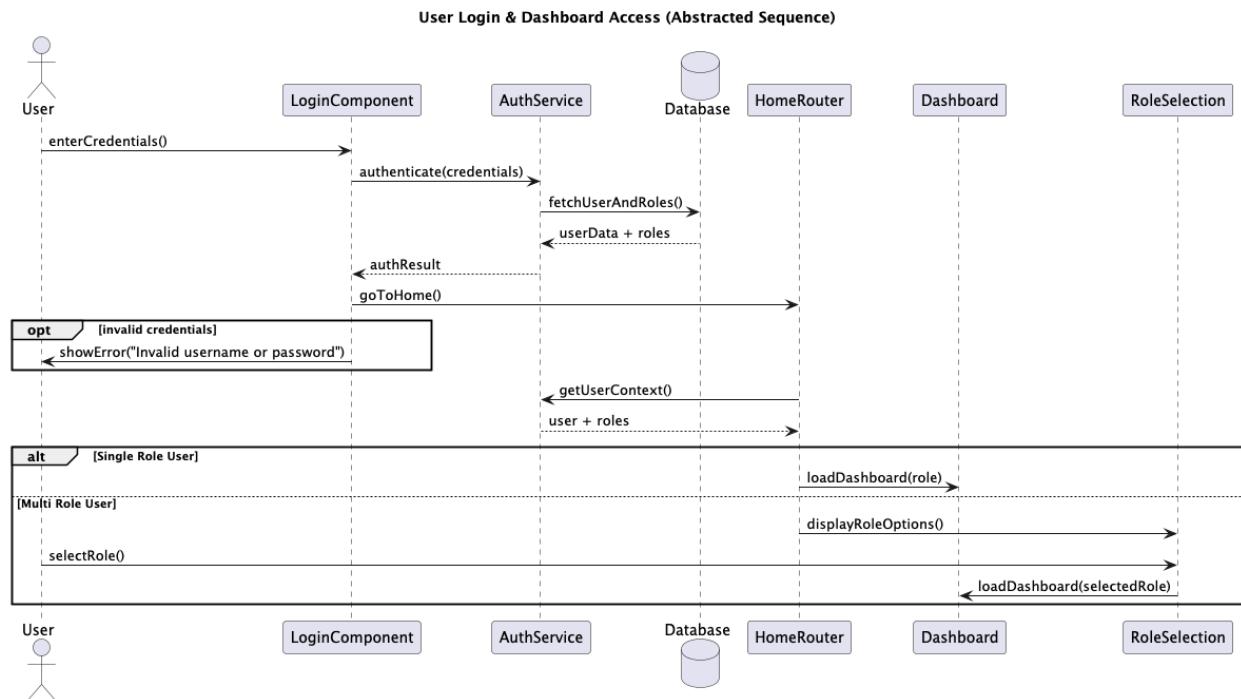


Figure 10: Sequence Diagram - User Login & Dashboard Access

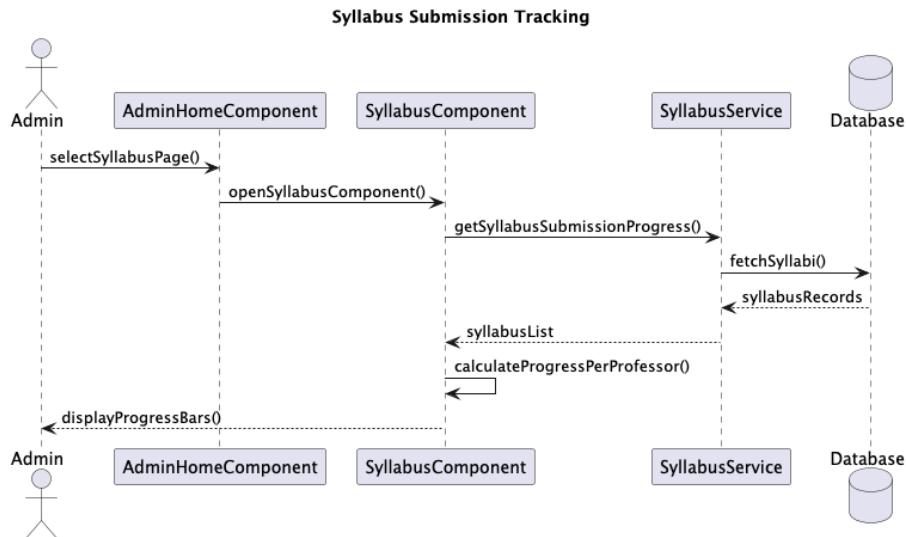


Figure 11: Sequence Diagram - Syllabus Submission Tracking

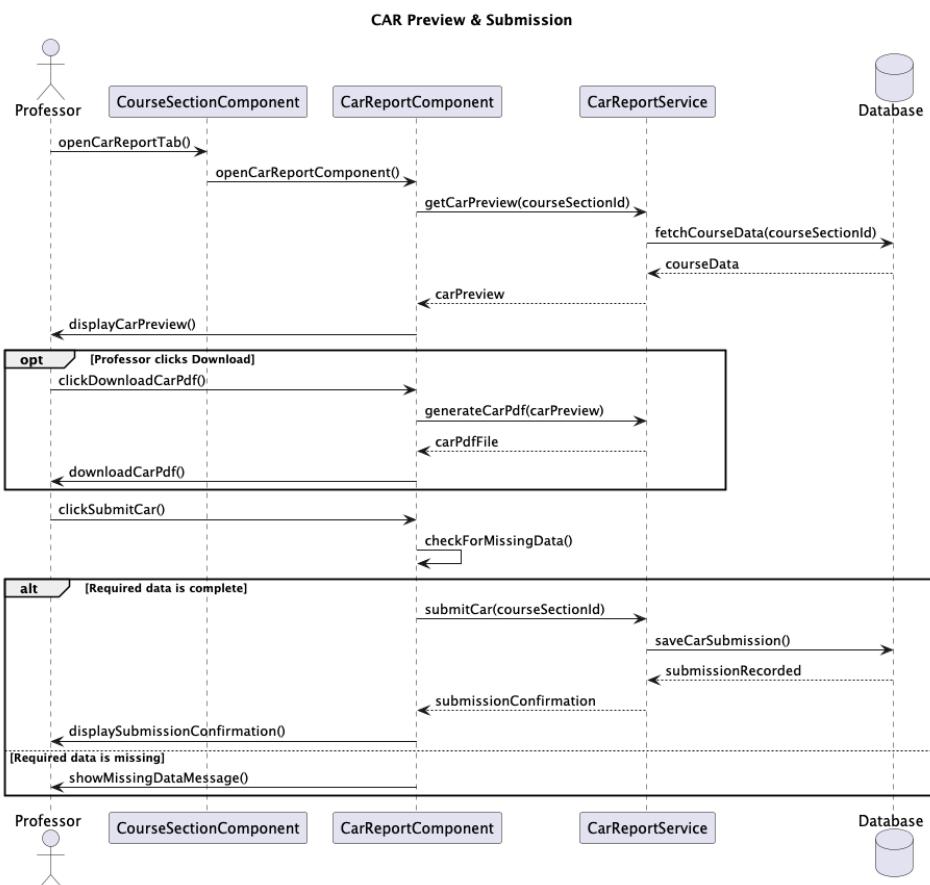


Figure 12: Sequence Diagram - CAR Preview & Submission

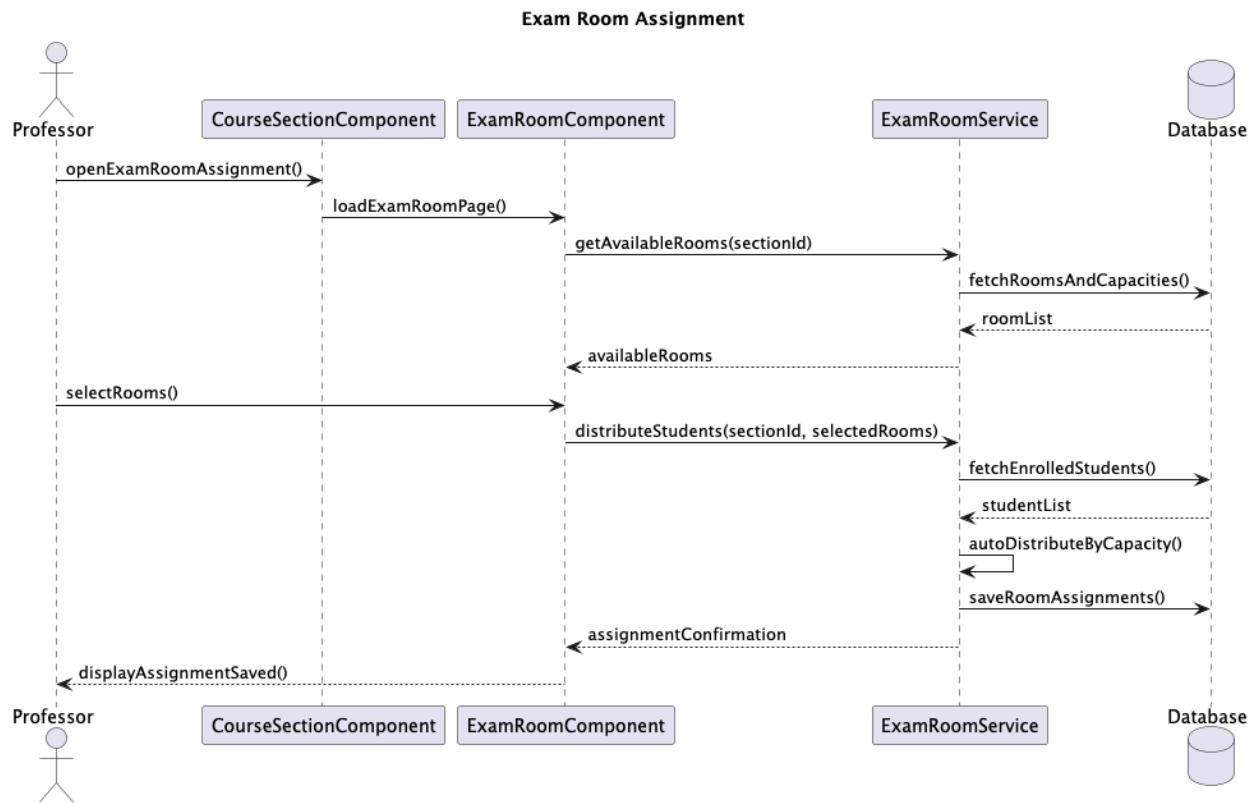


Figure 13: Sequence Diagram - Exam Room Assignment

2. *Activity Diagrams*: The activity diagrams represent the end-to-end workflow logic of selected user stories in CourseGenie+, showing user actions, system decisions, and process flow across different roles.

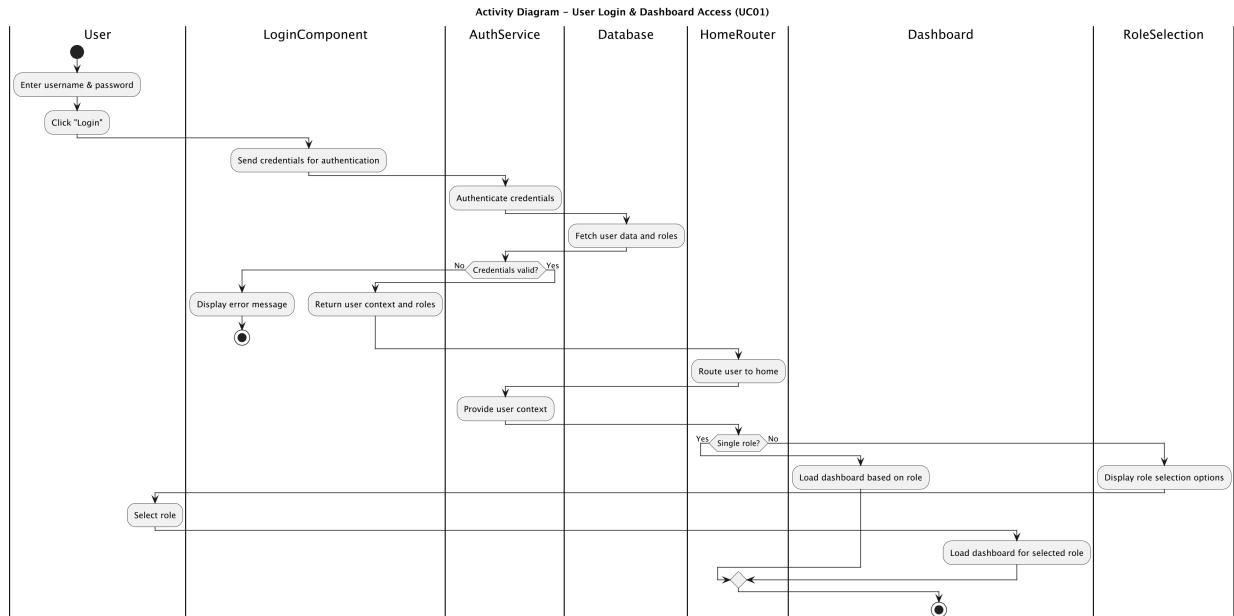


Figure 14: Activity Diagram - User Login & Dashboard Selection

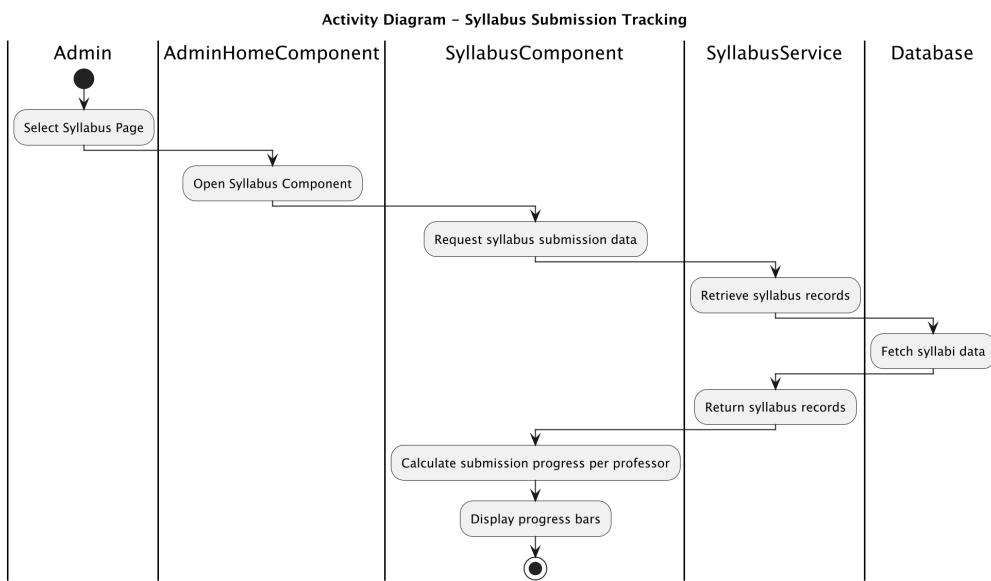


Figure 15: Activity Diagram - Syllabus Submission Tracking

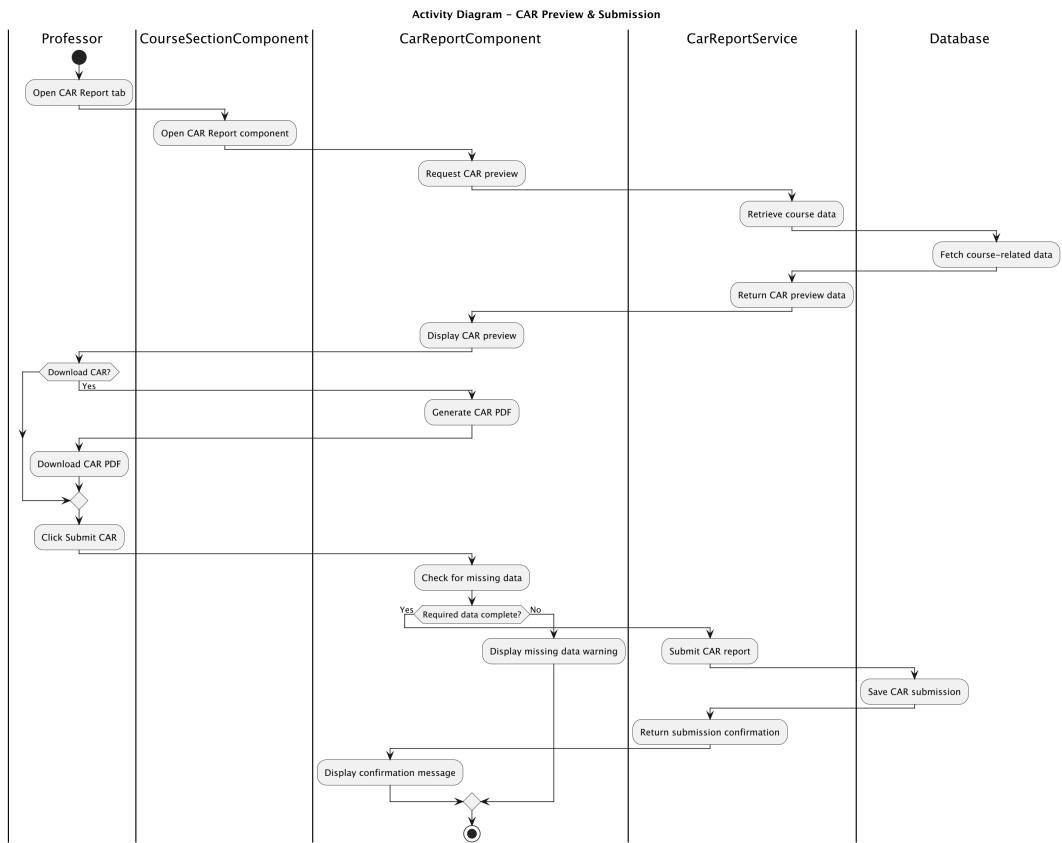


Figure 16: Activity Diagram - CAR Preview & Submission

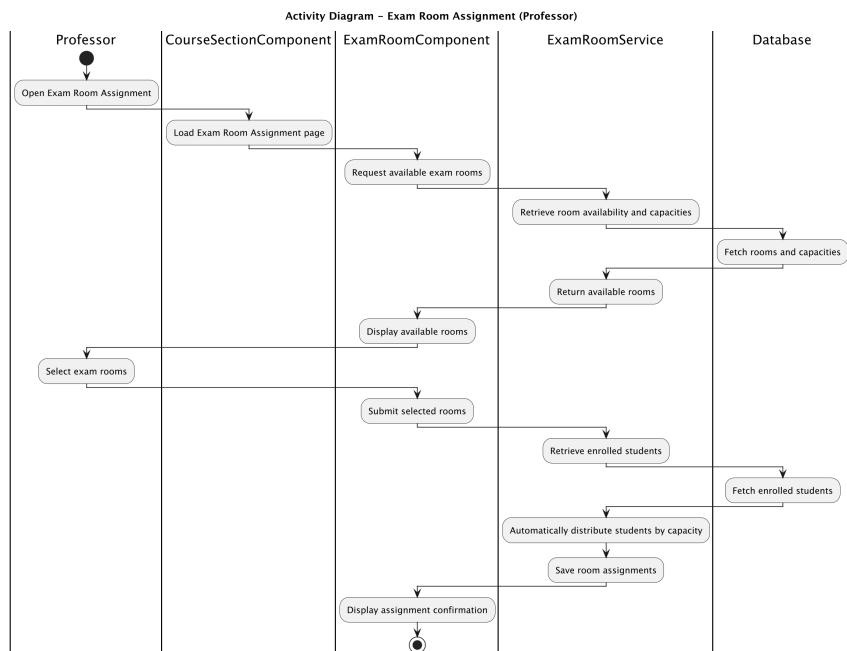


Figure 17: Activity Diagram - Exam Room Assignment

## XII. Configuration Requirements and Technologies

This section outlines the core technologies and configuration requirements used to implement and run CourseGenie+. It summarizes the frontend, backend, security, database, and deployment components required for the system to operate correctly.

Category	Technology / Tool	Configuration / Usage
Frontend Framework	Angular	Used to implement the client side user interface, dashboards, and role-based views for professors and administrators.
Backend Framework	Spring Boot	Used to implement backend services, business logic, and RESTful APIs following a layered MVC architecture.
Security Framework	Spring Security	Used to handle authentication, authorization, session management, and role-based access control.
Authentication Source	LDAP	Used as a centralized directory for user authentication and role retrieval, integrated through Spring Security.
Database Management System	MySQL	Used for persistent storage of users, courses, sections, assessments, CAR data, and exam related information.
API Communication	RESTful APIs	Used for communication between the Angular frontend and Spring Boot backend using HTTP methods.
Deployment Environment	Docker (Local)	Used for running the database and backend services in a containerized local development environment.

### XIII. Project Planning

This section outlines the project planning artifacts used to organize, schedule and assign responsibilities throughout the development of CourseGenie+. This includes a Work Breakdown Structure to define the major system components, a Gantt chart to visualize task scheduling and a RACI matrix to clarify task assignments among team members.

#### A. Work Breakdown Structure

The Work Breakdown Structure organizes the project into phases of a hierarchical structure showing the sub-systems on different levels.

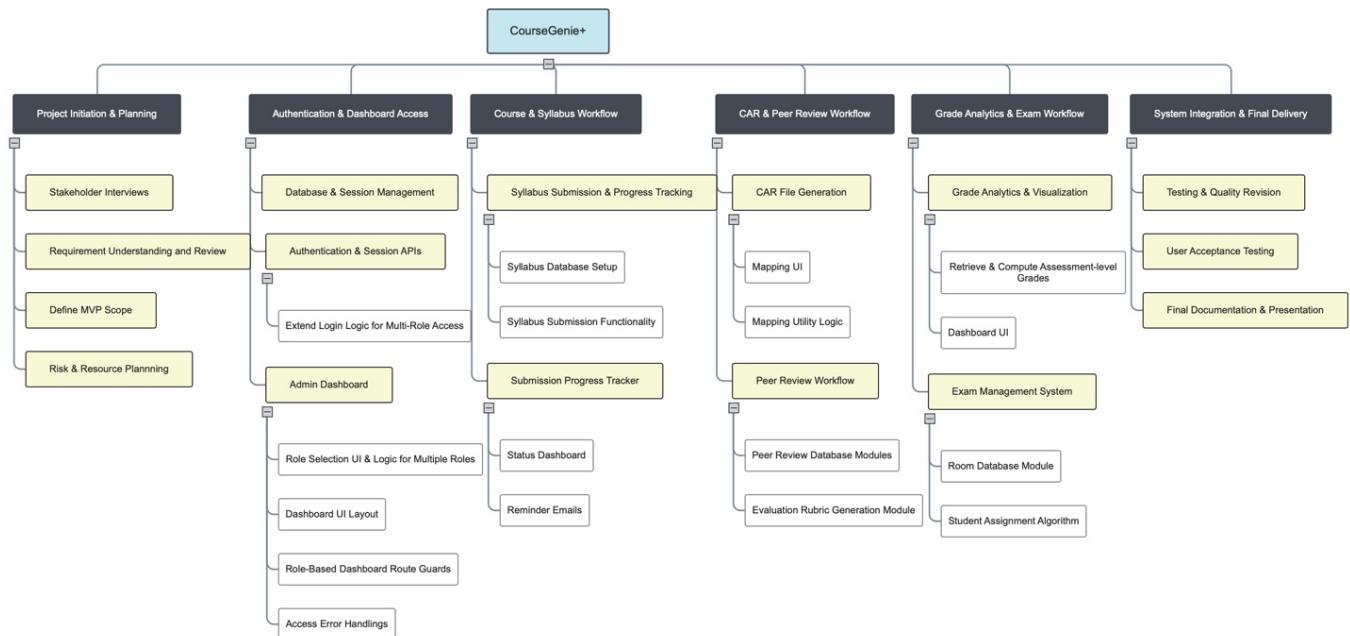


Figure 18: Work Breakdown Structure

## B. Gantt Chart

The Gantt Chart visualizes the task durations and sequencing during the planned timeline of the project completion.

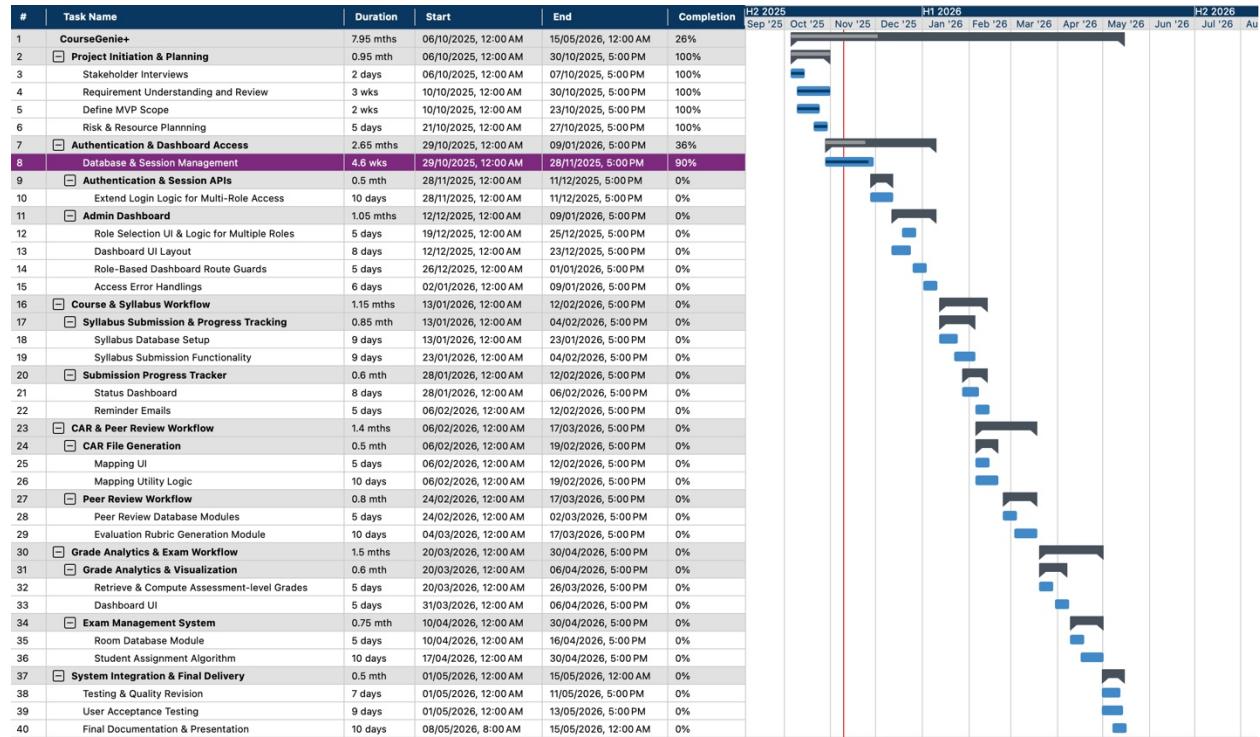


Figure 19: Gantt Chart

**1. Progress Reflection:** According to the previously defined Gantt chart, which represents the planned project timeline, the project is progressing as expected at this stage. The initiation, requirements elicitation, and risk and resource planning phases have been completed. Core system foundations, including authentication and database session management, have been implemented and incrementally tested as part of the agile development process. The Admin Dashboard functionality is currently in progress, with its main components developed and partially integrated. The next semester will primarily focus on continuing feature development, completing remaining modules, and further iterative integration in alignment with the original project plan.

### C. RACI Matrix

The RACI matrix clarifies who is responsible, accountable, consulted and/or informed of the main identified project tasks.

Task / Stakeholders	Jeeda Kotob Full-Stack Developer	Ryma Alt Tayeb Backend Developer	Ibraheem Mustafa Frontend Developer	Dr. Ali Assi Mentor	Mr.Qusai Hasan Co-mentor	Ms. Dazzil Castelino Institutional Effectiveness
Requirement Understanding and Review	R	I	I	A	C	C
Database & Session Management	A	R	I	C	I	I
Admin Dashboard	R	C	A	C	I	I
Syllabus Submission & Progress Tracking	A	R	I	C	C	C
CAR File Generation	A	R	I	C	C	C
Peer Review Workflow	I	R	A	C	C	C
Exam Management System	A	R	I	C	I	C
Rubric Generation Module	R	A	I	C	C	I
Testing & Quality Revision	R	I	A	C	C	I
Final Documentation & Presentation	A	R	I	C	C	I



Figure 20: RACI Matrix

## XIV. Risk Assessment

This section evaluates potential technical, operational and project-related risks that could affect the development and deployment of CourseGenie+. A risk matrix, interpretation legend and a risk register are used to identify, classify and plan to mitigate these risks.

### A. Risk Matrix

The 5x5 risk matrix below categorizes risks based on their impact on the project's outcomes in addition to their likelihood of occurring.

		Impact				
		1 (Insignificant)	2 (Minor)	3 (Noticeable)	4 (Major)	5 (Critical)
Likelihood	5 (Very High)	Medium	High	High	Critical	Critical
	4 (High)	Medium	Medium	High	High	Critical
	3 (Moderate)	Low	Medium	Medium	High	High
	2 (Low)	Low	Low	Medium	Medium	High
	1 (Very Low)	Negligible	Low	Low	Medium	Medium

Table 5: 5x5 Risk Assessment Matrix

## B. Legend

The legend defines how combined likelihood and impact scores translate into risk severity levels and the appropriate team response strategy.

Risk Level	Score Range	Meaning / Required Action
Low	1-7	Acceptable risk, no special intervention needed.
Medium	8-14	Needs monitoring during development.
High	15-19	Required planned mitigation and follow up in sprints to prevent delays and quality issues.
Critical	>=20	Must be addressed immediately, risks timeline or project success

Table 6: Risk Level Interpretation Legend

### C. Project Risk Register & Mitigation

The following risk register lists the risks identified to possibly occur during the development of CourseGenie+, along with their severity levels and individual mitigation strategies.

Risk	Type	Likelihood (1–5)	Impact (1–5)	Risk Score	Risk Level	Mitigation Strategy
Learning Curve in Spring Boot / Angular	Technical	4	4	16	High	Allocate time for learning Spring Boot and Angular and divide development tasks based on individual technical strengths.
Restricted access to MyCourses data/API	External / Integration	3	4	12	Medium	Design CourseGenie+ as an independent system with its own database, avoiding direct dependency on MyCourses APIs.
Incomplete Database Schema	Technical	3	4	12	Medium	Refine the database schema to align with requirements by updating entities, relationships, and attributes.

Frontend Stale Data	Technical	3	3	9	Medium	Implement frontend mechanisms to ensure views reflect the most recent backend state.
Evolving Requirements During Development	Project Management	5	3	15	High	Manage requirement changes through Scrum sprint planning, backlog prioritization, and regular mentor and stakeholder reviews.

Table 7: Project Risk Register & Mitigation Strategies

## XV. Resources and Budget Planning

Effective resource planning is essential to ensure that CourseGenie+ progresses smoothly across all development phases. Since the project is conducted within an academic context, resource allocation focuses on optimizing team expertise, available technology, and institutional support rather than financial assets. This section outlines the human, technical, and logistical resources required to successfully complete the project.

### A. Bill of Materials (BoM)

The Bill of Materials would identify all the necessary components, technologies, and resources that are required for the successful making and development of CourseGenie+. Since the project is mostly software, the BoM emphasizes hardware development and testing, as well as the major software frameworks and services that we are using.

Category	Component	Source	Purpose of Need	Estimated Cost
Software	Angular & Spring Boot Frameworks	Open Source	Frontend and backend development	AED 0 (Open Source)
Database	MySQL	Oracle	Data storage	AED 0 (Open Source)
Version Control	GitHub	GitHub	Repository management and collaborative development	AED 0 (Open Source)
Documentation and Design Tools	PlantUML (Open Source), MindView, Canva, SlideBazaar	Open Source / Paid Access	Project planning and system design diagrams	~ AED 100
Learning Resources	Online learning resources for UML design, frontend, and backend technologies	LinkedIn Learning	Skill development and framework understanding	~ AED 900
Unexpected Costs	Contingency expenses	Personal	Covers unforeseen software tools, subscriptions, or minor project expenses	~ AED 200
Total Estimated Cost				~ AED 1200

Table 8: BoM

## B. Budget Summary

The overall project budget is estimated at approximately AED 1200, with costs mainly associated with learning resources and contingency planning. Core development tools and frameworks are open source and do not include licensing fees. This budget allocation ensures that the project remains cost effective while supporting the required technical and academic objectives.

## XVI. Testing Plan

The test plan outlines the set of functional test cases designed to validate the core features for both professors and administrators. The purpose of this first-cut test plan is to verify that essential workflows are performed as expected. The cases below specify test conditions, inputs, execution steps, and expected outcomes to ensure consistency during testing. Test cases corresponding to implemented features were executed and validated, while remaining cases are included as planned tests for the future development phase. In addition to functional testing, JUnit-based unit tests are used to validate backend services and business logic.

ID	Title	Trace	Preconditions	Input	Steps	Expected Output	Actual Output	Pass/Fail
1	Login -> Professor Dashboard	US1, UC01	User exists with Professor role only	Username: axacad5 Password: ali123	Open Login -> Enter credentials -> Login	Redirects to Professor Dashboard	Redirects to Professor Dashboard	Pass
2	Login -> Admin Dashboard	US1, UC01	User exists with Admin role only	Username: dmccad Password: dezzil123	Open Login -> Enter credentials -> Login	Redirects to Admin Dashboard	Redirects to Admin Dashboard	Pass
3	Login -> Role	US1, UC01	User has Admin +	Username: kxkcad	Login	Role selection	Role selection	Pass

	Selection Popup		Professor roles	Password: khalid123		popup displayed	popup displayed	
4	Select Professor from Popup	US1, UC01	Role popup visible	Click Continue as Professor	Click Continue	Redirects to Professor Dashboard	Redirects to Professor Dashboard	Pass
5	Select Admin from Popup	US1, UC01	Role popup visible	Click Continue as Admin	Click Continue	Redirects to Admin Dashboard	Redirects to Admin Dashboard	Pass
6	Invalid Login Shows Error	UC01 AF2.1	Any user	Username: axacad5 Password: wrongPass	Attempt login	"Invalid username or password." displayed	"Invalid username or password." displayed	Pass
7	Admin Dashboard Displays Course Cards	US2	Logged in as Admin	---	Load dashboard	Course cards visible	Course cards visible	Pass
8	Open Course Section List	US2	Course cards visible	Click course card	Click card	Section cards displayed		
9	Navigate to Syllabus Page	US3	Logged in as Admin	Click Syllabus tab	Click tab	Redirect to Syllabus Page		
10	Syllabus Progress Bars Display	US3	On Syllabus Page	---	Load Page	Progress bars grouped by department		
11	View Professor Syllabus Submissions	US4	On Syllabus Page	Click professor name	Click name	Submission list displayed		
12	Preview Syllabus	US4	Submission list visible	Click syllabus entry	Click entry	Preview opens		
13	Download Syllabus PDF	US4	Preview visible	Click Download	Click Download	PDF downloaded		
14	Show No Submissions Case	US4	Professor has no submissions	Open professor record	Open record	"No submissions available" displayed		
15	Overdue Indicator Appears	US4	Submission past deadline	---	Open list	"Overdue" displayed		

16	Auto Reminder for Syllabus Due	US5	Due in 2 days	---	Wait for scheduler	System sends reminder email		
17	Add Section Button Visible	US6	On course page	---	Load page	"Add Section" visible		
18	Open Add Section Form	US6	Add Section visible	Click Add Section	Click button	Form opens		
19	Navigate to CAR Files Page	US7	Logged in as Admin	Click CAR Files tab	Click tab	Redirects to CAR Files Page		
20	CAR Progress Bars Display	US7	On CAR Files Page	---	Load page	Progress bars grouped by department		
21	View Professor CAR Submissions	US8	On CAR Files Page	Click professor name	Click name	Submission list displayed		
22	Preview CAR File	US8	CAR list visible	Click CAR entry	Click entry	CAR preview opens		
23	Download CAR PDF	US8	Preview visible	Click Download	Click Download	PDF downloaded		
24	Show No CAR Submissions	US8	Professor has none	Open record	Open record	"No submissions available" displayed		
25	Overdue CAR Marked	US8	Late CAR exists	---	Open list	"Overdue" displayed		
26	Auto Reminder for CAR Due	US9	Due in 2 days	---	Wait for scheduler	System sends reminder email		
27	Navigate to Peer Review Page	US10	Logged in as Admin	Click Peer Review tab	Click tab	Redirects to Peer Review Page		
28	Peer Review Progress Display	US10	On Peer Review Page	---	Load page	Progress bars grouped by department		

29	Generate CAR Preview	US12, UC02	Professor on section page	Open CAR tab	Open tab	CAR preview generated		
30	Submit CAR (Complete Data)	US12, UC02 AF4.3	All required data present	Click Submit	Click Submit	Submission recorded + confirmation		

Table 9: System Test Plan

## XVII. Ethics and Impact on Society

CourseGenie+ is designed with ethical considerations that prioritize security, privacy, and responsible data usage. Access to system features is strictly controlled through role-based authorization, ensuring that professors and administrators can only view and manage information relevant to their responsibilities. Academic data such as grades, assessment records, and peer reviews are treated as confidential and are accessible only to authorized users. The system incorporates LDAP-based authentication and follows a security aware design, with data encryption planned for deployment to further enhance protection. Additionally, CourseGenie+ limits data collection to essential academic information only, supporting responsible and ethical use of institutional data.

From a societal perspective, CourseGenie+ positively impacts academic environments by improving efficiency and consistency in higher education workflows. By automating tasks such as syllabus management, CAR generation, exam coordination, and administrative tracking, the system significantly reduces manual effort for faculty and administrators. Standardized system-generated academic artifacts promote consistency and reliability across departments, supporting accreditation and quality assurance processes. Overall, CourseGenie+ contributes to more effective academic administration, allowing educators to focus more on teaching and continuous improvement rather than repetitive administrative tasks.

## XVIII. Stakeholder Feedback, Reflection and Lessons Learned

Feedback from project stakeholders and the evaluation committee was generally positive. The committee highlighted the Syllabus Management and CAR automation as key strengths and differentiating features of CourseGenie+, noting their value in reducing manual academic workload and supporting accreditation processes. This feedback validated the project's core focus on faculty-centered workflow automation.

Stakeholders also recommended strengthening system security through mechanisms such as data encryption and multi-factor authentication, which have been identified as important enhancements for future development. Additionally, discussions highlighted that while some features exist in MyCourses, CourseGenie+ adds value by centralizing data and enabling advanced automation and analytics, rather than duplicating existing LMS functionality.

Through this phase, the team learned the importance of early architectural planning, security considerations, and continuous stakeholder feedback. Adopting an Agile approach enabled incremental development, early testing of implemented features, and flexibility in accommodating feedback while remaining aligned with the original project plan.

## XIX. Conclusion

This phase of the CourseGenie+ project resulted in the successful development of a functional prototype that addresses key gaps in both faculty and administrative academic workflows. Using an agile, iterative development approach has allowed the team to adapt to evolving requirements and incorporate stakeholder feedback throughout the semester.

While time has faced some constraints, planned mitigations were followed to overcome these constraints in a way without impacting the planned project timeline. The current system establishes a solid technical and architectural foundation, presenting the main modules of the system. CourseGenie+ provides a strong baseline for continued planned development, with future phases focusing on completing remaining modules, enhancing security features, and expanding system automation capabilities.

## REFERENCES

- [1] J. Sánchez, M. Serrano, A. Duarte and F. García, "Learning Management Systems for Higher Education: A Comparative Study," *Journal of Educational Technology*, vol. 21, no. 2, pp. 145-162, 2024.
- [2] A. Yamani, S. Alharthi and H. Smirani, "Blackboard vs. Brightspace: A Comparative Study," *International Journal of Emerging Technologies in Learning (iJET)*, vol. 17, no. 9, pp. 45-58, 2022.
- [3] B. Simon, K. Müller and R. Patel, "Adoption of Moodle Tools: Instructor and Student Perspectives," *Education and Information Technologies*, vol. 29, no. 4, pp. 2105-2122, 2024.
- [4] Q. Li, Y. Jung and A. F. Wise, "How instructors use learning analytics: The pivotal role of pedagogy," *Journal of Computing in Higher Education*, vol. 37, pp. 55-75, 2025.
- [5] M. Hernández-de-Menéndez, R. Morales-Menéndez, C. A. Escobar and R. A. Ramírez Mendoza, "Learning analytics: state of the art," *International Journal on Interactive Design and Manufacturing (IJIDeM)*, vol. 16, no. 3, pp. 1209-1230, 2022.
- [6] P. Ngulube and M. M. Ncube, "Leveraging Learning Analytics to Improve the User Experience of Learning Management Systems in Higher Education Institutions," *Information*, vol. 16, no. 5, pp. 419-419, 2025.
- [7] X. Kui, N. Liu, Q. Liu, J. Liu, X. Zeng and C. Zhang, "A survey of visual analytics techniques for online education," *Visual Informatics*, vol. 6, no. 7, 2022.

- [8] D. Vlachopoulos and A. Makri, "A systematic literature review on authentic assessment in higher education: Best practices for the development of 21st century skills, and policy considerations," *Studies in Educational Evaluation*, vol. 83, p. 101425–101425, 2024.
- [9] E. Jenkins, R. D'Aoust, S. de Souza Elias, H.-R. Han, P. Sharps and C. Alvarez, "Faculty peer review of teaching taskforce: A quantitative descriptive research study for the peer review process," *Nurse Education Today*, vol. 106, p. 105055, 2021.