B.Sc. (Hons) in Information Technology

IT2080 IT Project

Y2S2

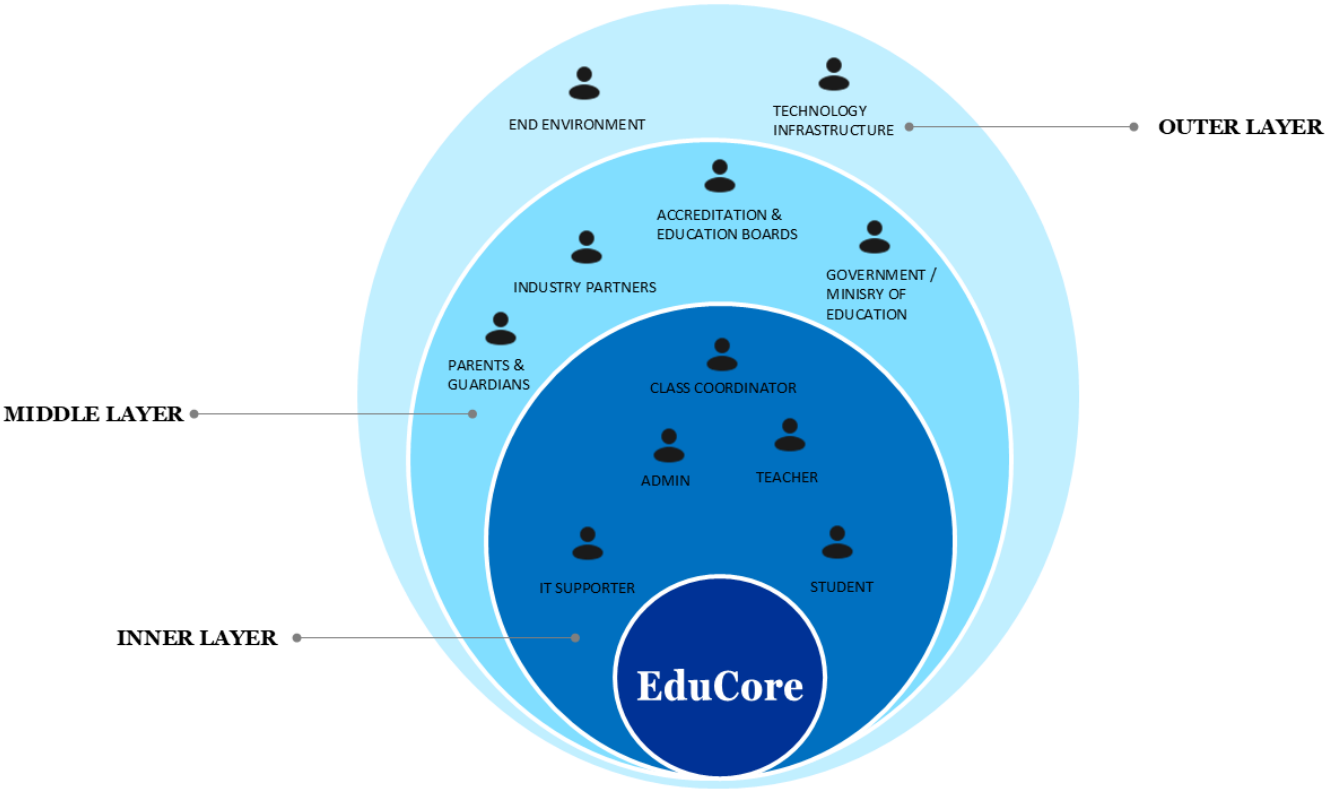
Activity 2

Project Group : ITP25_B5_110

IT Number	Student Name	Student E-mail Address	Contact Number
IT23623590	Dasanayaka D.M.P.M	it23623590@my.sliit.lk	0762758512
IT23650220	Obbris M.W.S	it23650220@my.sliit.lk	0703141361
IT23756182	Navodya H.S.N	it23756182@my.sliit.lk	0787850857
IT23604230	Nishshanka N.M.L.S	it23604230@my.sliit.lk	0713592726
IT23861718	Nethmi P.P.U	it23861718@my.sliit.lk	0741794363

Table of Contents

1. Onion Diagram



2. List the Functional Requirements for the direct system users (in the innermost layer of the Onion diagram or main stakeholders).

Functional Requirements

1. Student

- Students can register using secure password encryption and email verification.
- Lecture notes, slides, and recorded or live-stream lessons are all available to students.
- Students can upload assignments, take quizzes, and check the status of their submissions.
- Grades, finished assignments, and instructor comments are all visible to students.
- The system notifies users of upcoming tests, due dates, and grades.

2. Teacher

- Teachers use role-based access to safely log in.
- Lecture notes, videos, homework, and tests can all be posted by teachers.
- Materials' publication and expiration dates can be set by teachers.
- Teachers have the ability to create tests, assignments, and grade submissions.
- Assessment can receive individualized feedback and grades from teachers.
- Activity logs, grades, and attendance can all be tracked by teachers.
- Each student's performance summary can be exported by teachers.

4. Class Coordinator

- Class coordinator designates the time, date, place, instructor, course code and designation to determine the schedules.
- Class coordinators place the schedule of exam date on chapters with information on the date, time, and venue where they will be done.
- One-time arrangements on activities like seminars, workshops, and guest lectures is organized by class coordinator.

- Existing schedules are updated by Class coordinators who adjust date, time, place, or instructor assigned, and automatically notify students and staff.
- Outdated schedules or cancelled ones are finally deleted by the class coordinator confirming the deletion.

4. Admin

- Create, edit, disable, or delete user accounts.
- Assign specific roles (Student, Teacher, Examiner, IT Support, Admin).
- Assign students and teachers to specific classes or batches.
- Set notification preferences, manage permissions.
- Monitor and respond to feedback submitted by users.
- Produce academic, attendance, and activity reports in PDF or Excel.
- Track all system changes and actions taken by admins.

5. IT Support

- Examine, sort, and rank support requests.
- Modify the ticket's status to either Pending, In Progress, or Resolved.
- Address problems in the ticket system.
- Save the history of resolved tickets for evaluation and enhancement.

3. List the related NFRs and analyze them user wise.

Non-Functional Requirements

Stakeholder	Non-Functional Requirements	Description
Students	Usability	Simple and intuitive UI
	Performance	24/7 access
	Reliability	No data loss in submissions
Teachers	Maintainability	Ability to update content and grades without downtime.
	Compliance	Content uploads must meet academic integrity and copyright rules.
	Performance	Quizzes and materials must be published in 5 seconds after submission.
Class Coordinator	Scalability	The system must allow the coordinator to manage multiple classes and subjects without lag.
	Performance	Attendance and grade reports must generate in under 5 seconds for up to 500 students.
	Security	All student performance and attendance data must be encrypted in storage and during transmission.
Admins	Maintainability	For simple updates and fixes, the codebase should be modular.
	Scalability	Support for more stations/users
	Security	Strong password policies, role-based access
IT Support	Confidentiality	Support tickets containing sensitive user data must be encrypted
	Accessibility	IT staff should be able to respond to tickets from a mobile-friendly dashboard.

4. State the technical requirements for the system.

Technical Requirements

- Web-based System
- Backend - Node.js with Express.js
- Front-end - React.js
- Database - MongoDB
- Authentication - Email verification, password encryption
- Version Control - GitHub
- APIs - REST API

5. Model the requirements using a use case diagram.

6. Write down the use case descriptions for 5 main use cases in the diagram.

Use case 01: Manage User Accounts

Number	1	
Name	Manage User Accounts	
Summary	The administrator can create, edit, delete, or disable user accounts (Students, Teachers, Class Coordinator, IT supporters). The admin can also reset passwords and manage login details.	
Priority	1	
Preconditions	▪ Admin must be logged into the system.	
Postcondition	▪ User accounts are created, updated, disabled, or deleted. Users receive confirmations for account activation or changes.	
Primary Actor	Admin	
Secondary Actors	Students, Teachers, Class Coordinator, IT Support	
Trigger	The admin navigates to the "User Management" section.	
Main Scenario	**Step**	**Action**
	01	The admin logs into the system.
	02	The admin navigates to "User Management."
	03	The system displays a list of all registered users.
	04	The admin chooses to create, edit, delete, or disable a user account.

	05	If creating, the admin enters details (name, email, role, phone number, etc.).
	06	The system validates details and requests email/phone confirmation.
	07	If editing, the admin updates user details such as profile information, login credentials, or assigned roles.
	08	If deleting/disabling, the admin confirms the action.
	09	The system applies changes and updates the user's status.
	10	The system sends confirmation or error messages.
Extensions	**Step**	**Branching Action**
	6a	If email/phone number is invalid, system displays: "Invalid contact details. Please try again."
	6b	If account confirmation is pending, account remains inactive until verified.
	9a	If update fails due to technical error, system displays: "Unable to update account. Please try again later."
Open Issues	▪ Should deleted accounts be permanently removed or archived (soft delete)? ▪ How to handle duplicate accounts?	

Use case 02: Upload Course Material

Number	2
Name	Upload Course Material

Summary	This scenario enables a teacher to upload course materials (PDF, videos, slides, quizzes, and assignments) into the EduCore LMS. The system allows the teacher to set visibility, organize content, and schedule availability for students.	
Priority	5	
Preconditions	▪ The teacher must be signed up and logged into the EduCore LMS. ▪ The system must be accessible online. ▪ Teacher must have access rights to the respective courses.	
Postcondition	▪ Class material is successfully uploaded and stored in the database. ▪ The uploaded material is visible to students (based on teacher's visibility settings). ▪ A notification is generated for students about the new content.	
Primary Actor	Teacher	
Trigger	The student wants to attempt a scheduled online exam.	
Main Scenario	**Step**	**Action**
	1	Teacher logs into EduCore LMS.
	2	Teacher navigates to the **Dashboard**.
	3	Teacher selects **Course Management**.
	4	Teacher clicks on **Add New Content**.
	5	Teacher selects the type of content (PDF, Video, Slide, Quiz, Assignment).
	6	Teacher uploads the file or enters quiz/assignment details.
	7	Teacher submits the content upload request.
	8	The system validates the content and stores it in the database.
	9	Confirmation message appears with content ID and timestamp.
	10	Notification is sent to enrolled students about the new material.
Extensions	**Step**	**Branching Action**

	6a	If upload fails (e.g., unsupported format, large file), the system shows an error, and prompts retry.
	9a	If database connection is lost, the system queues for the upload until reconnection.
Open Issues	▪ Should teachers be allowed to upload multiple content files at once? ▪ Should system automatically archive expired content, or let teacher manage it manually?	

Use case 03: Attempt Online Exam

Number	3	
Name	Attempt Online Exam	
Summary	This scenario enables a student to view an online test that has been uploaded to EduCore by a teacher and complete the questions within the system prior to the due date.	
Priority	5	
Preconditions	▪ The student needs to be signed up and signed into the EduCore Learning Management System. ▪ The teacher must publish the test and make it accessible to the students' class. ▪ The system needs to be accessible online.	
Postcondition	▪ Responses from students are stored in the database. ▪ A confirmation message and submission ID are produced. ▪ The teacher receives a notification.	
Primary Actor	Student	
Trigger	The student wants to attempt a scheduled online exam.	
Main Scenario	**Step**	**Action**
	1	The student logs into EduCore LMS.
	2	The student selects the relevant course from the dashboard.
	3	The system displays the course page with upcoming exams.

	4	The student clicks on the scheduled exam.
	5	The system opens exam instructions and paper.
	6	The student reads the questions and types in answers for short/essay questions
	7	The student can navigate between questions and review answers.
	8	The student clicks the **"Submit Exam"** button when finished.
	9	The system validates that all mandatory questions are answered.
	10	The system stores the answers in the database.
	11	A confirmation page appears showing submission ID and timestamp.
	12	Notification is sent to the teacher about the submission.
Extensions	**Step**	**Branching Action**
	6a	If an internet connection is lost, auto-save answers periodically until reconnection.
	9a	If required questions are unanswered, show a warning before submission.
	8a	If the exam time expires, the system automatically submits saved answers.
Open Issues	▪ Would it be possible to download the question paper from the system for offline use? ▪ Is it appropriate for students to revise their responses after submitting them but before the due date?	

Use case 04 : Timetable & Scheduling Management

Number	04
Name	Timetable & Scheduling Management
Summary	This use case aims at allowing the class coordinator to create, update, and manage the timetables of classes as well as exams and events within EduCore LMS. These timetables can then be seen by students and teachers in real-time.

Priority	4	
Preconditions	▪ The class coordinator should be enrolled and enrolled to the EduCore LMS. ▪ Underlying courses, topics, and classes should be already in the system. ▪ System should be online.	
Postcondition	▪ The updated or new schedule is held in the database. ▪ Students and teachers receive notifications concerning the updates of the timetable. ▪ The new schedule is reflected, as shown in respective dashboards.	
Primary Actor	Class Coordinator	
Trigger	The class coordinator wants to create or update a schedule/ timetable.	
Main Scenario	**Step**	**Action**
	1	The EduCore LMS coordinates through login to EduCore LMS.
	2	The coordinator of the classes clicks on the timetable and scheduling module.
	3	There will be showing of available schedules and the possibility to create and update.
	4	The class coordinator clicks on the option either of either Create New Schedule or Update Schedule.

	5	The coordinator indicates time, date, venue, the course code, subject and a teacher.
	6	The coordinator can include instructions (e.g. bring lab coat).
	7	The timetable entry made is saved by the coordinator.
	8	The system confirms the conflict (e.g. overlapping rooms and teachers).
	9	In case it is genuine, the system saves the new schedule in the database.
	10	Students and teachers are alerted.
	11	Revised schedule is reported on the student and teacher dashboards.
Extensions	**Step**	**Branching Action**
	5a	If mandatory (date, time, course code) fields omission, the system displays an error.
	8a	If a conflict is detected (room or teacher double-booking),the system alerts the coordinator and suggests alternatives.
	7a	If internet is disconnected, then unsaved data are auto saved and will be saved once it is connected.
Open Issues	▪ Or are students allowed to make requests (e.g. rescheduling of clashes)?	

	▪ Will the schedule potentially export to PDF/ Excel, offline, to take instantly to a flight? ▪ Are notifications to be through emails, or SMS or both?

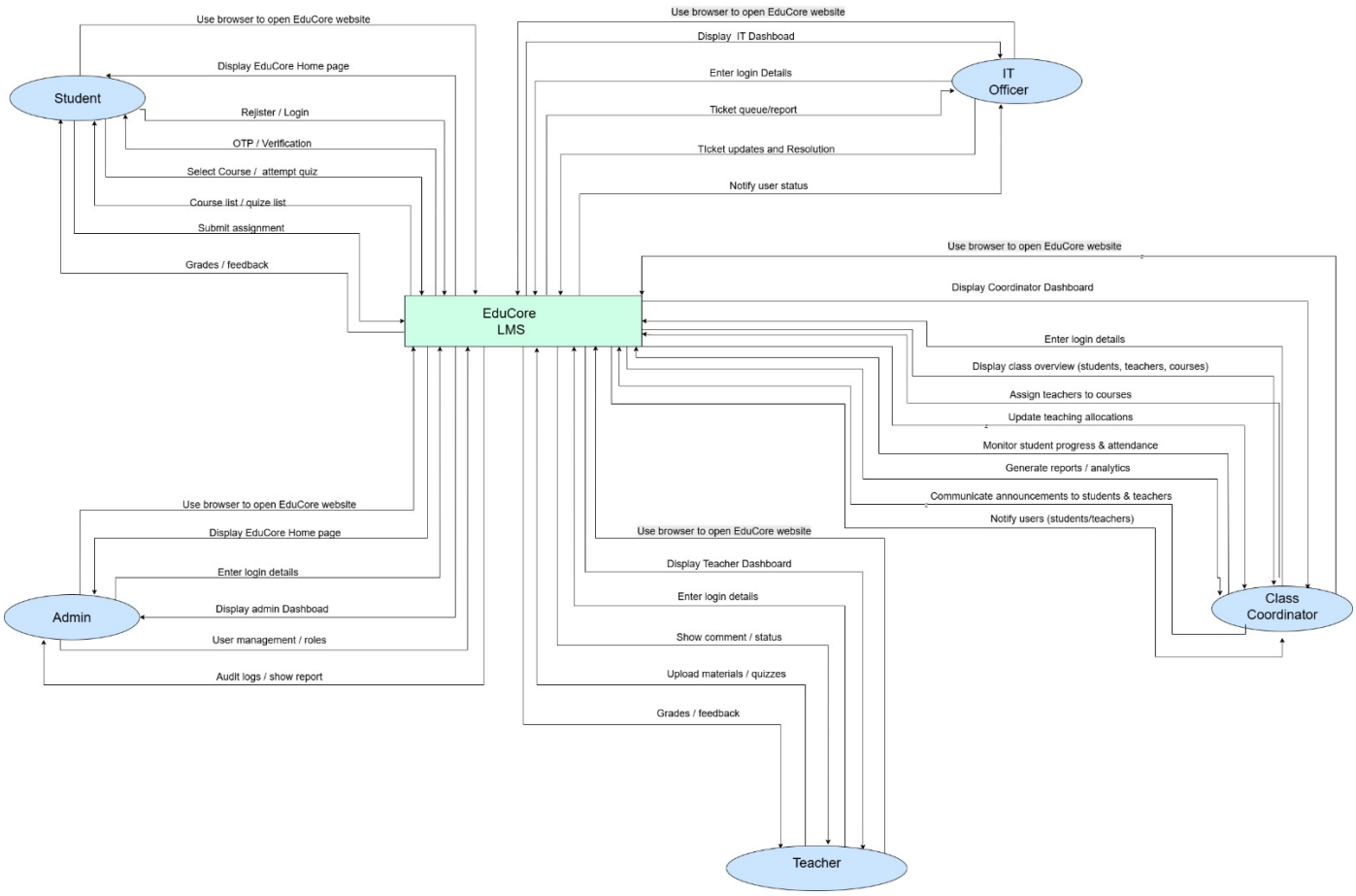
Use case 05 : Manage IT Support Ticket

Number	
Name	Manage IT Support Tickets
Summary	The IT Support Management module streamlines how technical issues are reported, tracked, and resolved. Users can easily submit tickets, and IT staff manage and update them through a central dashboard. users receive fast, reliable support. .
Priority	5
Preconditions	▪ The user is logged into the LMS. ▪ The support ticket submission form is accessible. ▪ The system is connected to the support database.
Postcondition	▪ The support ticket is successfully recorded in the system. ▪ The assigned IT staff is notified of the new ticket. ▪ The user receives confirmation of ticket submission with a reference ID.
Primary Actor	user

Trigger	A user encounters a technical issue or request and submits a support ticket through the LMS.	
Main Scenario	**Step**	**Action**
	1	A user logs in to the LMS.
	2	User navigates to the IT Support section.
	3	System assigns ticket a unique ID and logs it in the database.
	4	User submits a ticket describing the technical issue.
	5	IT Support Staff view tickets on the central dashboard.
	6	The staff informs the teacher about the problem and offers good solutions.
	7	Staff works on the issue and Staff update ticket status (e.g., Open, In Progress, Resolved).
	8	Staff send updates or resolutions to the user.
	9	User gets notified of ticket progress and resolution.
Extensions	**Step**	**Branching Action**

	6a	If the ticket details are invalid, the system will ask the user to correct the information.
	9a	If there is no response from the user, the system will automatically close the ticket after a specified time.
	8a	
Open Issues	▪ Handling duplicate tickets for the same problem. Improving automated notifications to ensure users don't miss updates. ▪ Reducing ticket resolution time for urgent issues. ▪	

7. Develop suitable diagrams to show visual presentation of data flow, the process. Flow and Data Connection to support the above (Eg: System Diagram, Flow chart, DFD)



8. Create a suitable plan to develop the project as a team.

Project planning as a team

The development of the EduCore is aimed at creating a user-friendly platform for students and Teachers regarding managing Class content in a Institute. Our team member gathered requirements for this platform by observing some Class web site and our Courseweb. From requirements collection and system design to development, testing, deployment, and maintenance, the plan addresses every stage of the project lifecycle.

Function	Member
User Management	Navodya H.S.N
Course content Management	Obbris M.W.S
Assessment Management	Dasanayaka D.M.P.M
Schedule Management	Nishshanka N.M.L.S
Support Management	Nethmi P.P.U

Our work plan.

Duration	Task	Team Members
Week 1-2	Planning, Requirement Gathering and Documentation	All
Week 3	Basic UI and setting up	All
Week 4-7	User management Function Development	Navodya H.S.N
	Course Content Function Development	Obbris M.W.S
	Assessment managing Function Development	Dasanayaka D.M.P.M
	Schedule Management Function Development	Nishshanka N.M.L.S
	Support Management Function Development	Nethmi P.P.U
Week 8-9	Testing and Documentation	All