

VIETNAM NATIONAL UNIVERSITY - HO CHI MINH CITY
UNIVERSITY OF INFORMATION TECHNOLOGY
FACULTY OF INFORMATION SYSTEMS



FINAL PROJECT REPORT
INFORMATION SYSTEM ANALYSIS AND DESIGN
ENGLISH CENTER MANAGEMENT SYSTEM

Ninh Thiên Bảo - 21520621

Lê Quốc Khánh - 21520283

Lê Gia Kiệt - 212522255

Nguyễn Thị Thủy - 25122662

Instructors: **PhD. Cao Thị Nhạn, M.S. Dương Phi Long**

Table of content

Chapter 1: INTRODUCTION	5
1.1. Introduction	5
1.2. Research problems / Problems of the project	5
1.3. Research Objectives / Objectives of the project	6
1.4. Scope of project / Limitations of the study	6
1.5 Development tools	7
Chapter 2: REQUIREMENT SPECIFICATION	8
2.1. Situation survey/ Survey of the current situation	8
2.1.1 Interview	8
2.1.2. Survey of the organization/ Current organizational situation	13
2.1.3. Survey of business processes, major activities	14
2.1.3.1. Board of Directors	14
2.1.3.2. Business Department	14
2.1.3.3. Accounting-finance department.....	15
2.1.3.4. Human resources department	16
2.1.3.5. Academic department	16
2.1.4. Survey of the existing system.....	17
2.1.4.1. Hardware	17
2.1.4.2. Software.....	17
2.1.4.3. Employee's skill	17
2.1.4.4. Number of employees	17
2.1.4.5. Frequency of students' attendance	17
2.2. Software requirements	18
2.2.1. Functional requirements	18
2.2.2. Non-functional requirements	18
Chapter 3: ANALYSIS AND DESIGN	20
3.1. Use-case Diagram.....	20
3.1.1 List of actors	24
3.1.2 List of Use-case	24

3.2. Use-case specification and activity diagram (Activity Diagram)	26
3.2.1. Login (For all users)	26
3.2.2. Manage Courses	29
3.2.2.1. Manage Chapters	33
3.2.2.3. Manage Classes.....	37
3.2.3.1. Manage Lessons.....	41
3.2.3.1.1. Manage Materials (For Teacher and Manager)	45
3.2.3.1.2. Manage Attendances (For Teacher and Manager)	48
3.2.4. Manage Classrooms	51
3.2.5. Manage Users	55
3.2.6. Export Class report	59
3.2.7. Analyze.....	60
3.2.8. Logout (For all users).....	62
3.2.9. Explore Courses (For User)	63
3.2.9.1. Explore Chapters (For User)	65
3.2.10. View Schedule (For User).....	66
3.2.11. Manage Classes (For Teacher)	67
3.2.11.1. Manage Lessons (For Teacher).....	69
3.2.12. View Classrooms	70
3.2.13. View Classes.....	72
3.2.13.1. View Lessons.....	74
3.2.13.1.1. View Materials	75
3.3. Sequence Diagram and Class Diagram	78
3.3.1. Login	78
3.3.2. Manage Courses	79
3.3.3. Manage Chapters.....	83
3.3.4. Manage	87
3.3.5. Manage Lessons	91
3.3.6. Manage Materials	95
3.3.7. Manage Attendances	99
3.3.8. Manage Classrooms	100

3.3.9. Manage Users	104
3.3.10. Export Class Report	108
3.3.11. Analyze	109
3.3.12. Logout	110
3.3.13. Explore Courses	111
3.3.14. Explore Chapters	112
3.3.15. View Schedule	113
3.3.16. Manage Classes (for Teacher)	114
3.3.17. Manage Lessons (for Teacher)	115
3.3.18. View Classrooms.....	116
3.3.19. View Classes.....	117
3.3.20. View Lessons.....	118
3.3.21. View Materials	119
3.4. Class Diagram and Entity class diagram	120
3.4.1. Overview Class diagram.....	120
3.4.2. Entity Class diagram.....	121
3.4.3. Class diagram specification	122
3.4.3.1. Course	122
3.4.3.2. Chapter	123
3.4.3.3. Class.....	124
3.4.3.4. Lesson	126
3.4.3.5. Material	127
3.4.3.6. Attendance	128
3.4.3.7. Room.....	129
3.4.3.8. Student	130
3.4.3.9. User.....	132
3.4.3.10. Login Controller.....	133
3.4.3.11. Course Controller	134
3.4.3.12. Chapter Controller	135
3.4.3.13. Class Controller	136
3.4.3.14. Lesson Controller	137

3.4.3.15. Material Controller	138
3.4.3.16. Attendance Controller	139
3.4.3.17. Classroom Controller	140
3.4.3.18. User Controller	141
3.4.3.19. Dashboard Controller	142
3.4.3.20. Schedule Controller	143
3.4.3.21. Homepage Controller	143
3.5. State Diagram	144
3.5.1. User	144
3.5.2. Class	145
3.5.3. Classroom	146
3.6. Mapping to Relational Model from Entity class diagram	147
Chapter 4: IMPLEMENTATION AND TESTING.....	148
4.1. Introduction to the implementation environment.....	148
4.2. User Interface.....	148
4.2.1. Login	148
4.2.2. Description about functions of manager	149
4.2.3. Description about functions of teacher	159
4.2.4. Description about functions of student	165
Chapter 5: CONCLUSION	168
5.1. Project achievement	168
5.2. Limitations	168
5.3. Development direction	169

Chapter 1: INTRODUCTION

1.1. Introduction

English is the key factor for global integration so there comes an increasing demand for learning the language. Many English centers were built in order to satisfy the needs of English learners. However, the biggest problem with most English centers is management. Most English centers manage their staff and students manually or by an unspecialized application, the management process will smooth if the number of staff and students is small. But in the long run, the process will be much slower and there will be some problems like: slow searching, limited functionality, time-consuming, data errors, etcetera which will cause a decline in the financial returns of those English centers.

With the aim to solve those long running problems of English centers management, our application was created. It is an application solely built for managing English centers, with easy-to-use UI, various useful functionality. It will shorten the searching process, enhance student experience, improve communication between teacher and students and most importantly is to make manage more efficient.

1.2. Research problems / Problems of the project

Most English centers apply technology into their management works. However, normal desktop management applications (Excel for example) doesn't allow remote collaboration for other teachers and administrators, it is very vulnerable to security breaches if not properly protected and would put students and teachers information at risk. Most tasks are not automated in normal management applications and it would be very time-consuming for businesses to do it step-by-step. If your English center grows, you may find it increasingly difficult to manage all of the data using only applications like Excel and this can lead to a lack of organization, communication difficulties, and other issues. Last but not least, normal applications don't allow storage of study documents, that means the only way for students

to finish their course work or just study is either to go to offline class or get a copy of the document.

The best way to resolve these problems is to have an application that can automate all these processes, with user-friendly UI and with multiple functionality suitable for students and teachers.

1.3. Research Objectives / Objectives of the project

Our application was created in order to achieve faster searching for large databases, and improve online communication between teachers and students. Here are some of our app's main features:

- Management of enrollment and placement test.
- Student/Teacher management.
- User management.
- Academic management.
- Classroom management.
- Academic results management.
- Revenue, expenditure and finance management.
- Analysis/report.
- Data recovery and Backup.

1.4. Scope of project / Limitations of the study

Due to the lack of time, human and expenses, we couldn't finish all of the features above the objective of our projects. Here are some of our main features that we spent the most time on:

- Assign user roles to the system.
- Class, lesson management.

- User management.
- Course, chapter management.
- Generate schedule.
- Attendance tracking.
- Classroom management.
- Analysis/Report.

1.5 Development tools

- Visual Studio Code/ IntelliJ IDEA as code editor.
- Java as the main programming language.
- Oracle database as database management system.
- Figma for UI designing.

Chapter 2: REQUIREMENT SPECIFICATION

2.1. Situation survey/ Survey of the current situation

2.1.1 Interview

General Report				
System: IELTS Fighter				
Owner: Nguyen Duc Anh		Created Day: 5/5/2023		
Order	Topic	Requirements	Start	End
1	Courses management	To understand about how the English center manages courses information and the flow of works	5/5/2023	5/5/2023
2	Students management	To understand about how the students are managed	5/5/2023	5/5/2023
3	Classes management	To understand how the English center can manage class	5/5/2023	5/5/2023
4	Study room management	To get informations about how study room are managed by the English center	5/5/2023	5/5/2023

Survey form	
Interviewee: Ninh Thien Bao	Interviewer: Nguyen Duc Anh
Place: IELTS Fighter	Start: 5/5/2023 End: 5/5/2023
Target: To understand how all processes work in an English center	Notes:
Interview details: <ul style="list-style-type: none"> – Introduction – System overview – Overview of the interview: <ul style="list-style-type: none"> ▪ Topic 1: Courses management ▪ Topic 2: Students management ▪ Topic 3: Class management ▪ Topic 4: Study room management 	Estimated time: <ul style="list-style-type: none"> – Introduction: 5 minutes – System Overview: 10 minutes – Overview of the interview: 40 minutes <p>Sum: 55 minutes</p>
General observation	
Unexpected occurrence	

Interviewee: Ninh Thien Bao	Created day: 5/5/2023	
Questions	Records	
Topic 1	How do you assess the English proficiency levels of students before placing them in a particular course?	Our English center distributes courses into three levels: beginner, intermediate, advanced. New students have to take an entrance test and we will put them in their suitable level
	Can you explain the functionality for managing course materials and resources within the course management system?	The materials of a course will be assigned by the Manager and Teacher for each lesson.
	Can you provide information about the system's capability to handle course materials and resources, such as textbooks, digital content, or multimedia resources?	Each class contains a certain course, and each course contains different materials, which students of that course can easily access and download.

Topic 2	How does the student management function handle student enrollment and registration processes?	After a new student completes the enrollment process regarding paperwork, center will provide an account for the student and assign a class level based on their proficiency
	What measures are in place to ensure data security and privacy within the student management system, particularly regarding student information?	For now, only allow authorized personnel to access the app and make sure that they have been trained on the security procedures and policies of the center.
	How does the student management function handle student enrollment and registration processes, and what information is required from students during the registration process?	After extracting information from registration forms, the app just needs to display the student's name and ID number for management purposes.

Topic 3	Can you describe the process of managing class sizes and ensuring an optimal student-to-instructor ratio for effective learning?	The class size will range from 30 to 50 students and will be managed by a teacher assigned by the English center.
	Can you explain how the class management function handles the organization and management of multiple class sections or levels within the English center system?	The class levels will be immediately classified in the course management section, meaning the same course will be labeled as beginner, intermediate, or advanced.
	Can you explain how the class management function handles the tracking and management of class progress, including materials and attendances?	As for class materials, they will be distributed in each lesson, besides, students will be able to check in through the app function during the class lesson.
Topic 4	What happens if two classes have the same schedule in the same room?	As every class is managed by the administrator, the application needs to give an error notification immediately when there is a registration clash, and

		prevent the registration of the conflicting class in the system.
	Can you explain how the study room management function handles the tracking and management of study room reservations during peak times or high-demand periods?	During peak times, reservations will be managed through separate procedures at the office, and the manager will update the system accordingly.

2.1.2. Survey of the organization/ Current organizational situation



Figure 1. 1. Current organizational situation

2.1.3. Survey of business processes, major activities

2.1.3.1. Board of Directors

Boards of Directors business: Collaborate with the Board to establish and monitor the center's financial policies and procedures. Work closely with the Board to develop a strategic plan that outlines the center's goals, objectives, and growth strategies. These plans should align with the mission and vision of the center and provide a roadmap for future development.

Boards of directors works process:

- Setting organizational goals and strategies
- Conducting board meetings
- Making important decisions
- Oversight of financial performance

2.1.3.2. Business Department

Student management business: The student management business of an English center involves various activities and processes to ensure a positive and productive experience for students, including student enrollment, student database management.

Student enrollment managing process:

- Handling inquiries
- Providing information about programs
- Assisting with application and registration

Student database management process:

- Inserting, updating student personal information, enrollment details
- Tracking student performance
- Tracking attendance records of students

Marketing business: Involves activities and strategies implemented to promote the center's program, services, and brand to attract new students.

English center's marketing and promotion process:

- Market research
- Creating a strong brand identity for the English center
- Digital marketing
- Public relations
- Organizing events, workshops, webinars...
- Customer Relationship Management

2.1.3.3. Accounting-finance department

Accounting-finance department management business: The accounting and finance department management in an English center involves overseeing financial operations, maintaining accurate records, and ensuring the center's financial health.

Accounting and financial management process:

- Financial reporting
- Expense control
- Revenue management
- Financial compliance
- Financial planning and analysis
- Cash flow management

- Auditing and controls
- Financial forecasting
- Financial decision-making

2.1.3.4. Human resources department

Human resources department management business: The human resources department management process in an English center involves managing employees to support the center's workforce effectively.

Employee management process:

- Overseeing recruitment
- Employee development
- Performance management
- Compliance
- Fostering a positive work culture

2.1.3.5. Academic department

Academic department management business: The academic department involves working around managing documents used for teaching and providing an effective learning environment for students.

Teaching document management process:

- Creating documents for teaching purposes
- Organizing documents
- Updating documents

Providing an effective learning environment for students:

- Fixing issues based on feedback from students.

2.1.4. Survey of the existing system

2.1.4.1. Hardware

- At the English Center, there's a total of 5 desktops equipped with core i7 processor, with 8GBs of RAM and with teachers' PC, laptop with good performance
- The computers in the center are connected to the LAN and the Internet via fiber optic cable.

2.1.4.2. Software

- The desktops will all have Windows 11 or MacOS 13 operating systems to support maximum performance for management software.

2.1.4.3. Employee's skill

- All teachers have basic office information skills; they can use the software relatively proficiently after being guided on how to use it.
- Managers are highly educated, capable of operating, managing, controlling different situations, making reasonable plans, and predicting risks for the center.

2.1.4.4. Number of employees

- The number of employees in the center includes 31 people.

2.1.4.5. Frequency of students' attendance

- The total number of registered students is 1000, the number of classes per day will fall between 500 and 600.

2.2. Software requirements

2.2.1. Functional requirements

- Storage function:
 - Account: AccountId, AccountName, Password, role ,...
 - Student: StudentID, UserID, ClassID.
 - Room: RoomId, Name, capacity,...
 - Course: CourseID, name, description, course_level....
 - Chapter: ChapterId, courseID, name, category, description,...
 - Classes: ClassesID, teacherID, name, courseID, date_start, date_end,...
 - Lesson: LessonID, ClassID, LearnDate, ChapterID,..
 - Class Attendance: LessonID, StudentID.
 - Material: MaterialD, LessonID, Path,...
- Searching function:
 - Get information about classes, rooms, courses, users, chapters of each course, lessons of each class.
- Statistical, report:
 - Generate graphs about attendance rate for each lesson in a class and popular courses based on enrollments.
 - Extract reports containing information of classes such as: classID, class name, course name, teacher name, List of Lessons.

2.2.2. Non-functional requirements

- Usability: The application should be easy to learn, navigate, and use for all types of users, regardless of their technical background or language skills.
- Reliability: The application should be available and functional at all times, with minimal downtime, errors, or crashes.

- Performance: The application should respond quickly and efficiently to user requests and tasks, even under heavy loads or network conditions.
- Security: The application should ensure the confidentiality, integrity, and availability of user data and transactions, and prevent unauthorized access, modification, or theft.
- Maintainability: The application should be easy to maintain, update, and troubleshoot, with clear and concise documentation, coding standards, and version control practices.

Chapter 3: ANALYSIS AND DESIGN

x3.1. Use-case Diagram

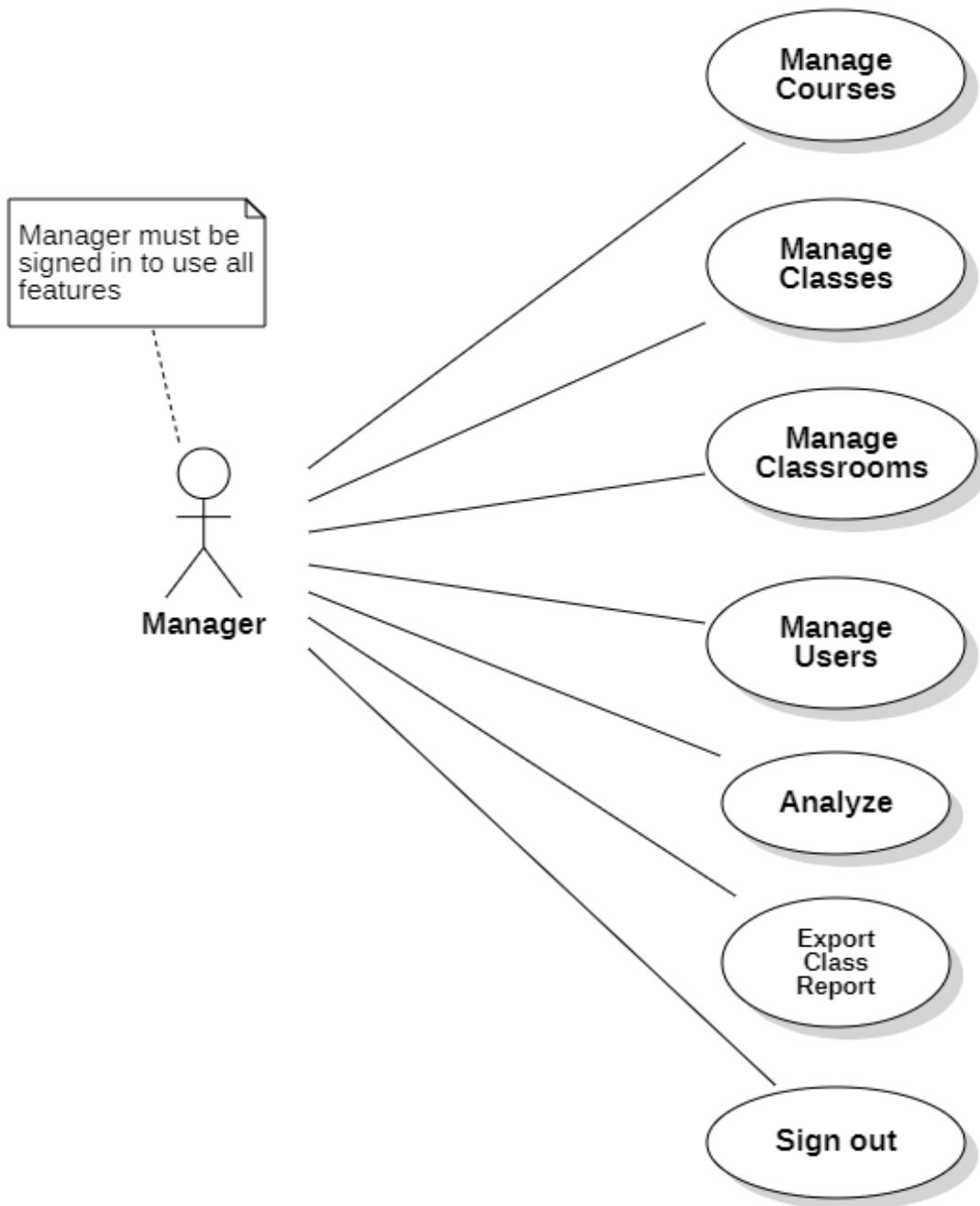


Figure 3. 1.Use case diagram for Manager in general

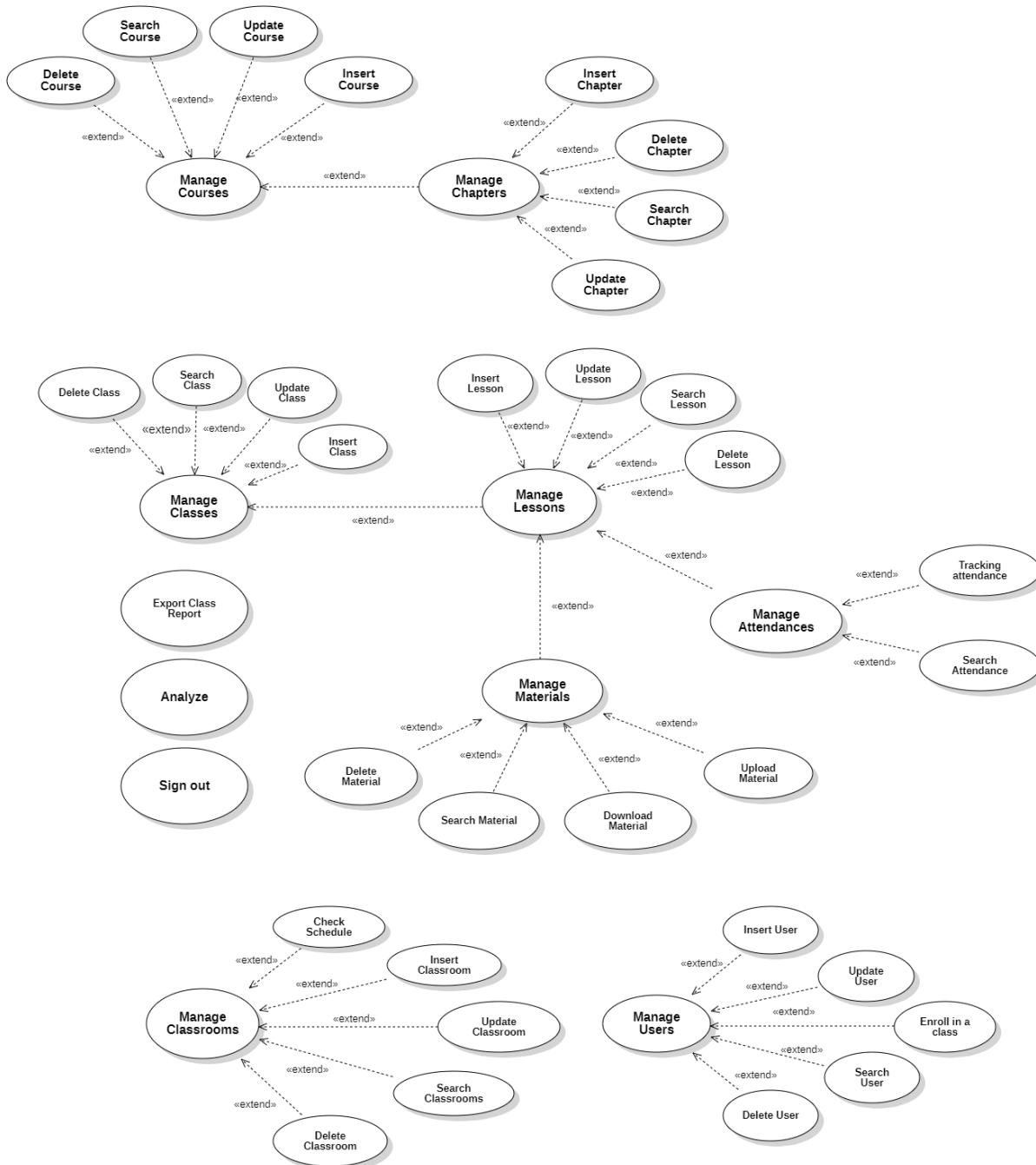


Figure 3. 2.Use case diagram for Manager in general

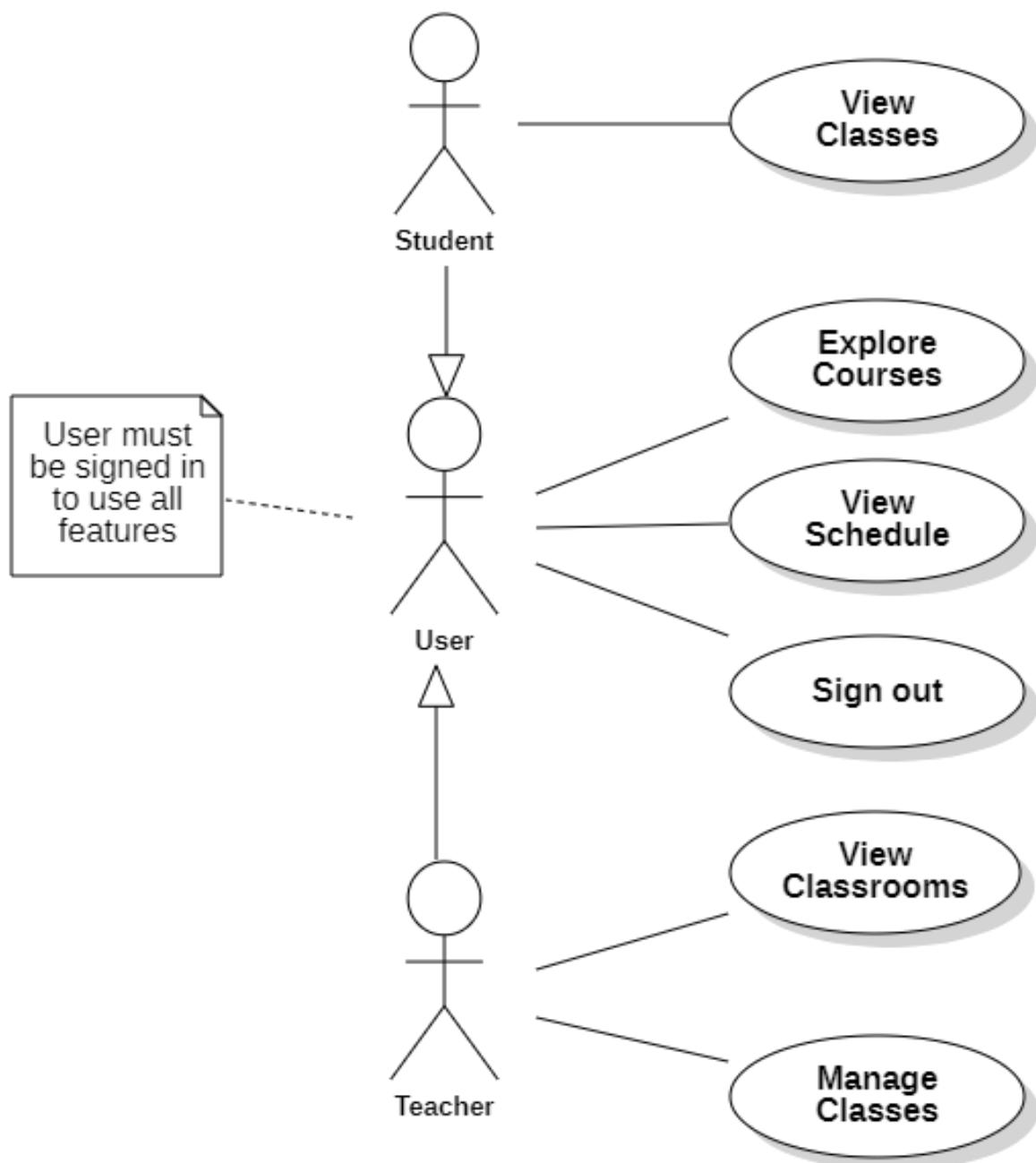
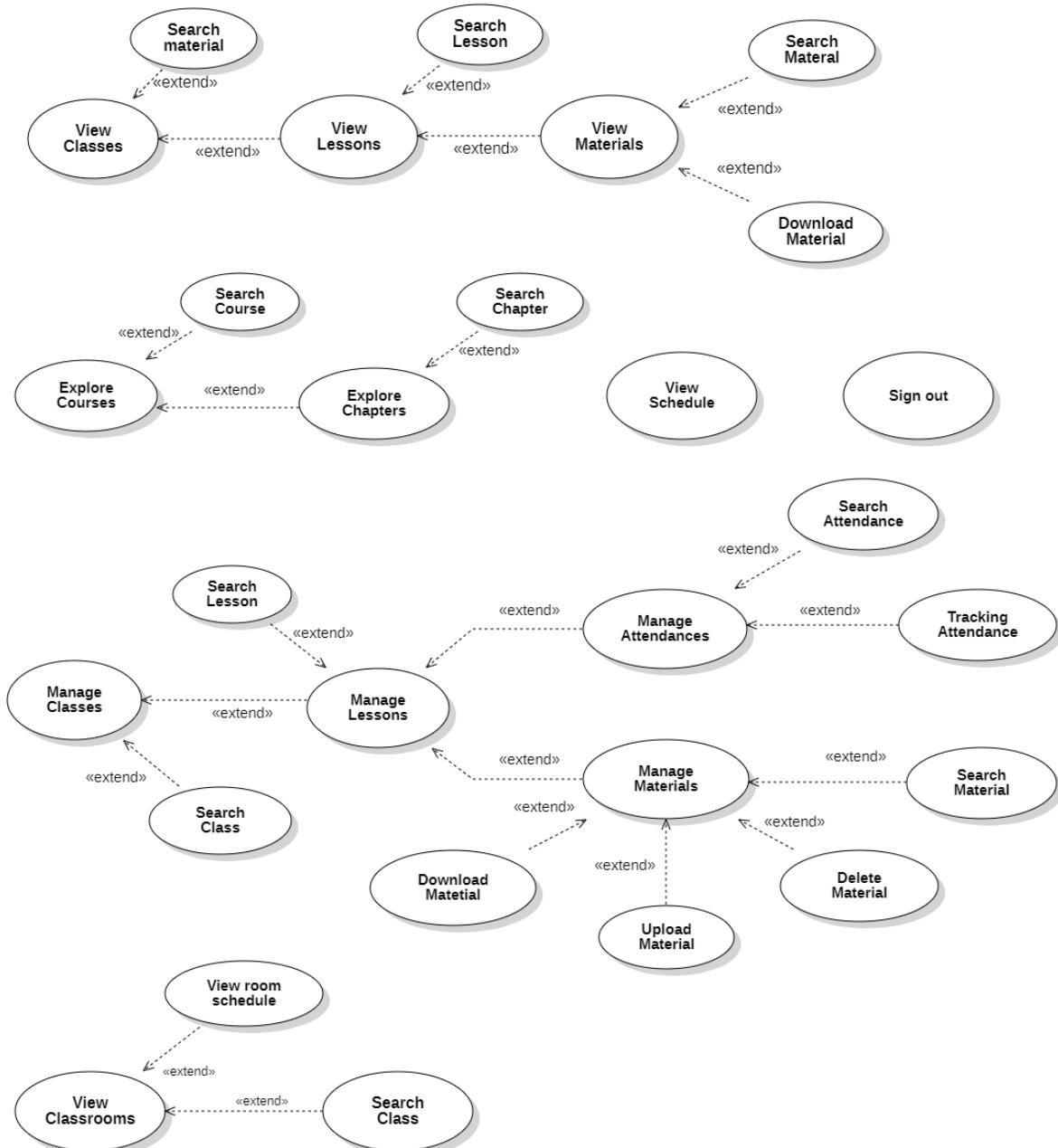


Figure 3. 3.Use case diagram for User in general

Figure 3. 4. Use case diagram for **User in details**

3.1.1 List of actors

Index	Actors		Meaning
1	Manager		People who manage the system
2	User	Teacher	People who use the system to manage tasks related to teaching
3		Student	People who use the system to tracking learning progress

3.1.2 List of Use-case

Role	Use-case	Meaning
Manager	1. Login (For all users)	Log in for using all features
	2. Manage Courses	Insert/Update/Delete Courses
	2.1. Manage Chapters	Insert/Update/Delete Chapters in a Course
	3. Manage Classes	Insert/Update/Delete Classes
	3.1. Manage Lessons	Insert/Update/Delete Lessons in a Class
	3.1.1. Manage Materials	Upload/Download/Search/Delete Material
	3.1.2. Manage Attendances	Search/Tracking Attendance students

	4. Manage Classrooms	Insert/Update/Delete/Search and View Schedule Classrooms
	5. Manage Users	Insert/Update/Delete/Search Users and Enroll in a class
	6. Export class report	Export class report about Class attendances
	7. Analyze	Analyze the performance of courses and classes
	8. Sign out (For all users)	Sign out of system
User	9. Explore Courses	Explore Course information.
	9.1. Explore Chapters	Explore all Chapters information in a Course
	10. View Schedule	View User's schedule
	11. Manage Classes	View all Classes that the Teacher is currently participating in.
	11.1. Manage Lessons	View all the designated lessons in a specific Class
	11.1.1. Manage Materials	Upload/Download/Search/Delete Material
	11.1.2. Manage Attendances	Search/Tracking Attendance students
	12. View Classrooms	View Classroom's status, schedule
	13. View Classes	View the classes that the student has participated in

	13.1. View Lessons	View the lessons taught in the class.
	13.1.1. View Materials	Search/Download the attached materials for each lesson.

3.2. Use-case specification and activity diagram (Activity Diagram)

3.2.1. Login (For all users)

Use-case name	Login
Description	User must log in to the system with the account provided by the Admin in order to use the system's features.
Trigger	Users click to button "Log in"
Pre-condition	User must have a valid account in the system.
Post-condition	<ul style="list-style-type: none"> – User receives a notification about the successful login. – User is granted access to the system and can perform actions according to their role and permissions.
Basic flow	<ol style="list-style-type: none"> 1. The System displays the login page with input fields for the user's username and password. 2. The user enters their username/email and password and clicks on the "Log In" button. 3. The system validates email & password in correct format. 4. The system verifies the user's credentials in database and grants access to the system. 5. The system displays successful login notification and redirects

	the user to homepage.
Alternative flow	<p>3a. If the user enters incorrect credentials:</p> <ul style="list-style-type: none"> – System displays an error message and prompts the user to re-enter their credentials. <p>Use-case continue at step 2.</p>
Exception flow	

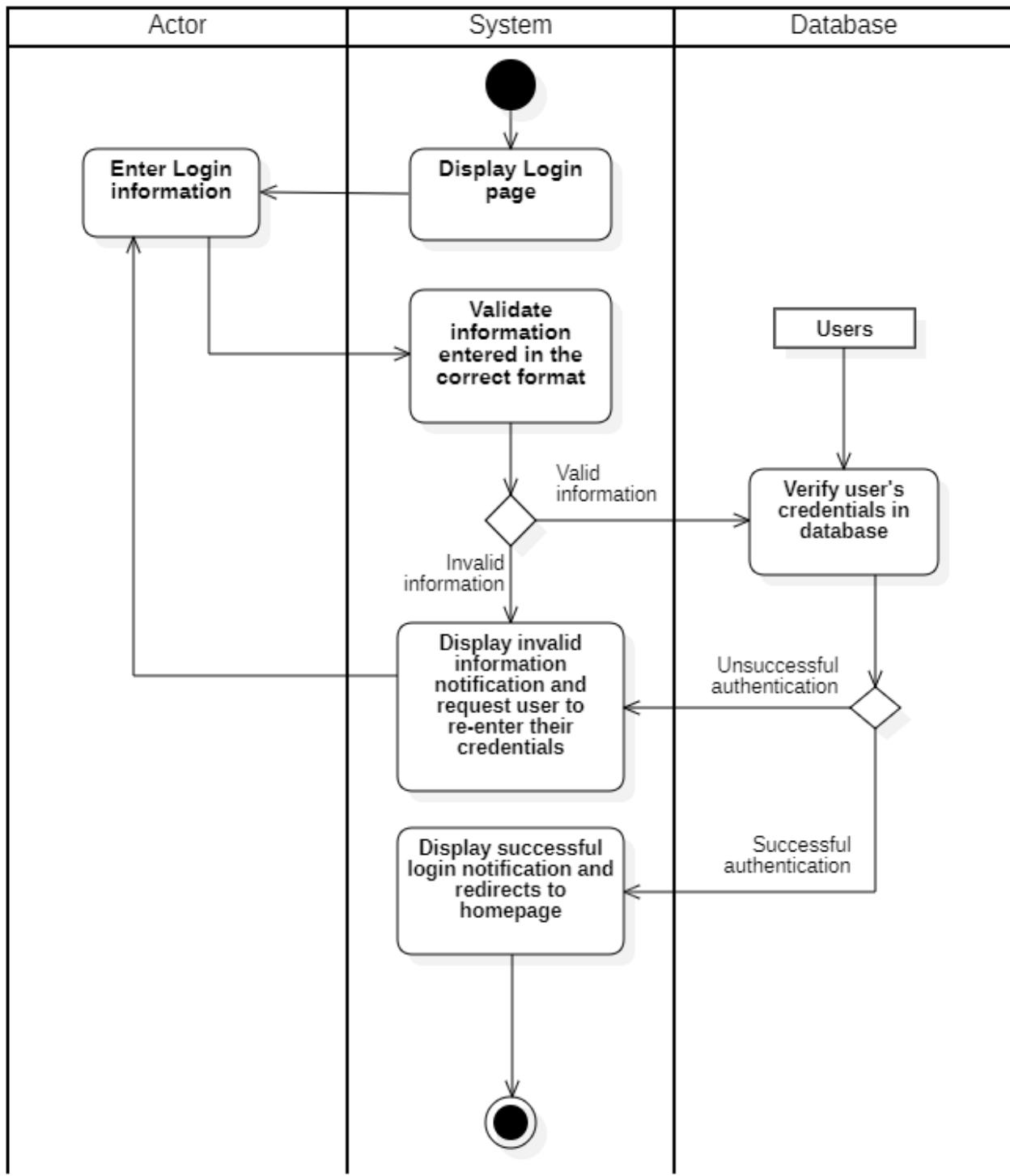


Figure 3. 5. Activity Diagram for Use case “Login”

3.2.2. Manage Courses

Use-case name	Manage Courses
Description	Manager can insert, update, delete and search Course information
Trigger	Manager clicks to “Courses” menu at homepage
Pre-condition	Manager’s status is signed in
Post-condition	<ul style="list-style-type: none"> – Course’s information successful updated in database. – Data displayed on the Course management page must be updated with the latest data.
Basic flow	<ol style="list-style-type: none"> 1. System displays courses management page 2. Manager chooses one function: insert/update/delete/search course’s information <ul style="list-style-type: none"> • Insert 3. System displays course information page 4. Manager enter course informations & submit 5. System validates course information in correct format 6. System validates course information with existed data in database 7. System stores course information in database 8. System displays successful notification and displays Courses management page <ul style="list-style-type: none"> • Update 9. Manager chooses one course to be updated 10. System displays course information page 11. Manager enter course informations & submit 12. System validates course information in correct format 13. System validates course information with existed data in database

	<p>14. System stores course information in database</p> <p>15. System display successful notification and displays Courses management page</p> <ul style="list-style-type: none"> • <i>Delete</i> <p>16. Choose course need to be delete</p> <p>17. Display confirm dialog then manager Confirm delete action</p> <p>18. If Manager confirm delete, System process delete request. Else, cancel action.</p> <p>19. System validates no external data required course information and selected item existed in database</p> <p>20. System delete course information from database</p> <p>21. System displays successful notification and display course management page with the latest data.</p> <ul style="list-style-type: none"> • <i>Search</i> <p>22. Manager select search function and enter keywords.</p> <p>23. System processes searching data</p> <p>24. System queries data base on given keywords</p> <p>25. System filtered data at course management page (data related to keyword is retained)</p>
Alternative flow	<p>5a. System display invalid info due to wrong format. Use-case continue at step 4.</p> <p>6a. System display invalid info because data had been existed in the database. Use-case continue at step 4.</p> <p>12a. System display invalid info due to wrong format. Use-case continue at step 11.</p> <p>13a. System display invalid info because data had been existed in the</p>

	database. Use-case continue at step 11.
Exception flow	

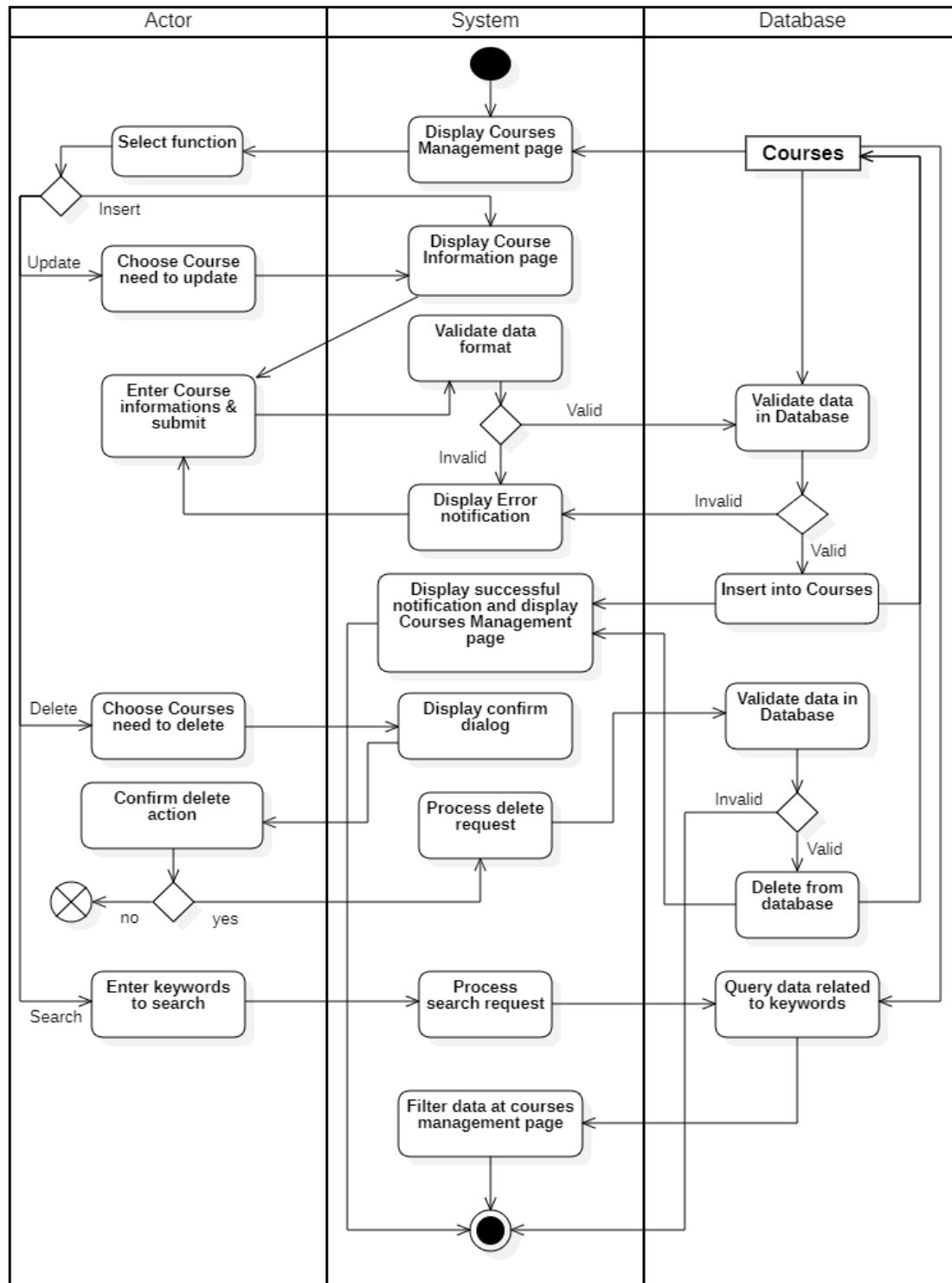


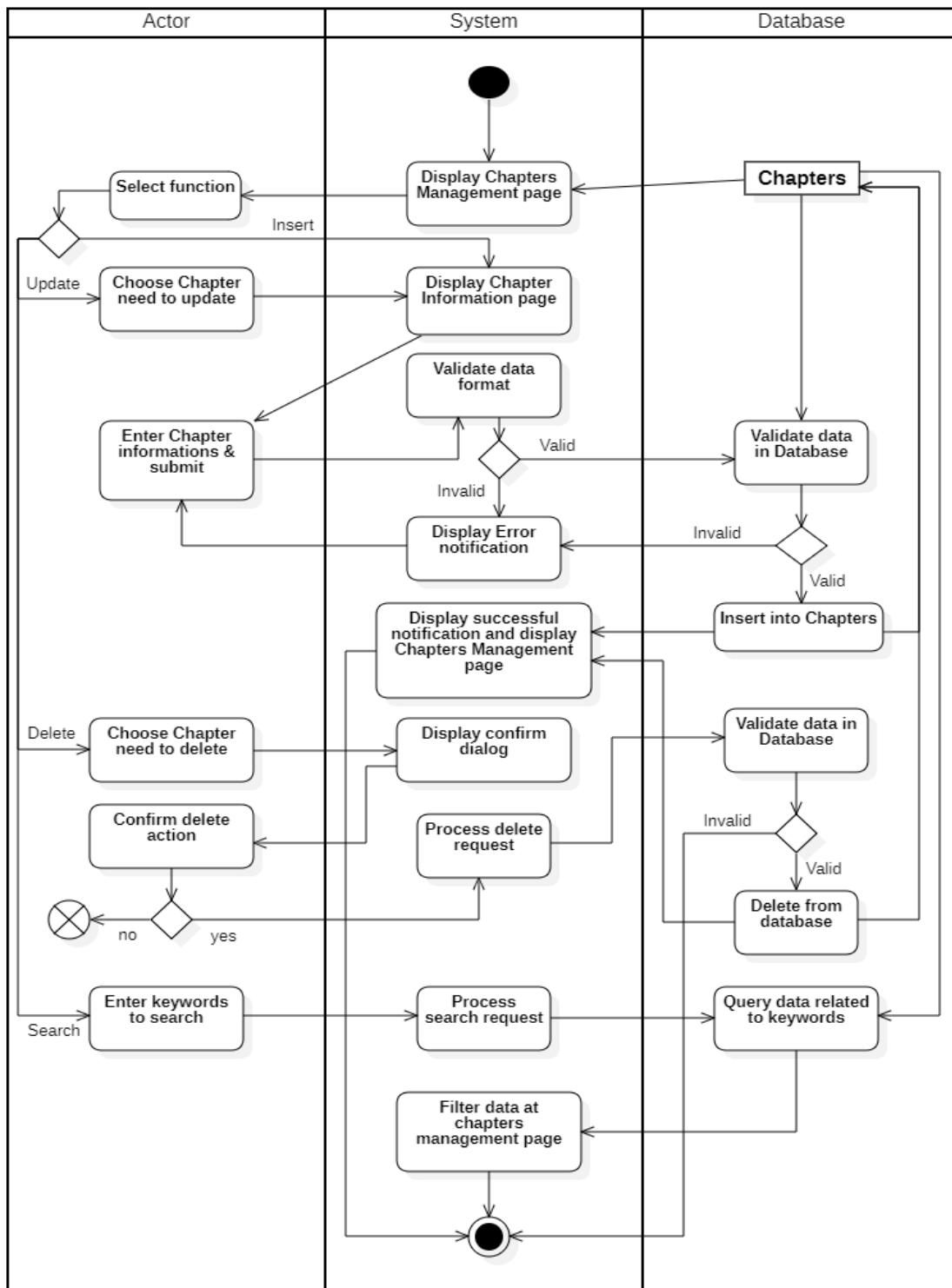
Figure 3. 6. Activity Diagram for Use case “Manage Courses”

3.2.2.1. Manage Chapters

Use-case name	Manage Chapters
Description	Manager can insert, update, delete and search Chapter information In a specific course
Trigger	Manager select a Course in Courses management page then clicks to “Manage Chapters”
Pre-condition	<ul style="list-style-type: none"> – Manager’s status is signed in – Manager has previously accessed Course management menu and selected a specific course.
Post-condition	<ul style="list-style-type: none"> – Chapters information successful updated in database. – Data displayed on the Chapters management page must be updated with the latest data
Basic flow	<ol style="list-style-type: none"> 1. Manager Select a Course and choose manage chapters 2. System displays chapters management page 3. Manager chooses one function: insert/update/delete/search chapter’s information <ul style="list-style-type: none"> • Insert 4. System displays chapter information page 5. Manager enter chapter informations & submit 6. System validates chapter information in correct format 7. System validates chapter information with existed data in database 8. System stores chapter information in database 9. System display successful notification and displays Chapters management page <ul style="list-style-type: none"> • Update

	<p>10. Manager chooses one chapter to be updated</p> <p>11. System displays chapter information page</p> <p>12. Manager enter chapter informations & submit</p> <p>13. System validates chapter information in correct format</p> <p>14. System validates chapter information with existed data in database</p> <p>15. System stores chapter information in database</p> <p>16. System display successful notification and displays chapters management page</p> <ul style="list-style-type: none"> • Delete <p>17. Choose chapter need to be delete</p> <p>18. Display confirm dialog then manager Confirm delete action</p> <p>19. If Manager confirm delete, System process delete request. Else, cancel action.</p> <p>20. System validates no external data required chapter information and selected item existed in database</p> <p>21. System delete chapter information from database</p> <p>22. System displays successful notification and display chapters management page with the latest data.</p> <ul style="list-style-type: none"> • Search <p>23. Manager select search function and enter keywords.</p> <p>24. System processes searching data</p> <p>25. System queries data base on given keywords</p> <p>26. System filtered data at course management page (data related to keyword is retained)</p>
Alternative	6a. System display invalid info due to wrong format. Use-case continue

flow	at step 5. 7a. System display invalid info because data had been existed in the database. Use-case continue at step 5. 13a. System display invalid info due to wrong format. Use-case continue at step 12. 14a. System display invalid info because data had been existed in the database. Use-case continue at step 12.
Exception flow	

Figure 3. 7. Activity Diagram for Use case “**Manage Chapters**”

3.2.3. Manage Classes

Use-case name	Manage Classes
Description	Manager can insert, update, delete and search Classes information
Trigger	Manager clicks to “Classes” menu at homepage
Pre-condition	Manager’s status is signed in
Post-condition	<ul style="list-style-type: none"> – Classes information successfully updated in database. – Data displayed on the Classes management page must be updated with the latest data
Basic flow	<ol style="list-style-type: none"> 1. System displays classes management page 2. Manager chooses one function: insert/update/delete/search class’s information <ul style="list-style-type: none"> • Insert 3. System displays class information page 4. Manager enter class informations & submit 5. System validates class information in correct format 6. System validates class information with existed data in database 7. System stores class information in database 8. System displays successful notification and displays Classes management page <ul style="list-style-type: none"> • Update 9. Manager chooses one class to be updated 10. System displays class information page 11. Manager enter class informations & submit 12. System validates class information in correct format 13. System validates class information with existed data in database

	<p>14. System stores class information in database</p> <p>15. System display successful notification and displays Classes management page</p> <ul style="list-style-type: none"> • <i>Delete</i> <p>16. Choose class need to be delete</p> <p>17. Display confirm dialog then manager Confirm delete action</p> <p>18. If Manager confirm delete, System process delete request. Else, cancel action.</p> <p>19. System validates no external data required class information and selected item existed in database</p> <p>20. System delete class information from database</p> <p>21. System displays successful notification and display Classes management page with the latest data.</p> <ul style="list-style-type: none"> • <i>Search</i> <p>22. Manager select search function and enter keywords.</p> <p>23. System processes searching data</p> <p>24. System queries data base on given keywords</p> <p>25. System filtered data at course management page (data related to keyword is retained)</p>
Alternative flow	<p>5a. System display invalid info due to wrong format. Use-case continue at step 4.</p> <p>6a. System display invalid info because data had been existed in the database. Use-case continue at step 4.</p> <p>12a. System display invalid info due to wrong format. Use-case continue at step 11.</p>

	13a. System display invalid info because data had been existed in the database. Use-case continue at step 11.
Exception flow	

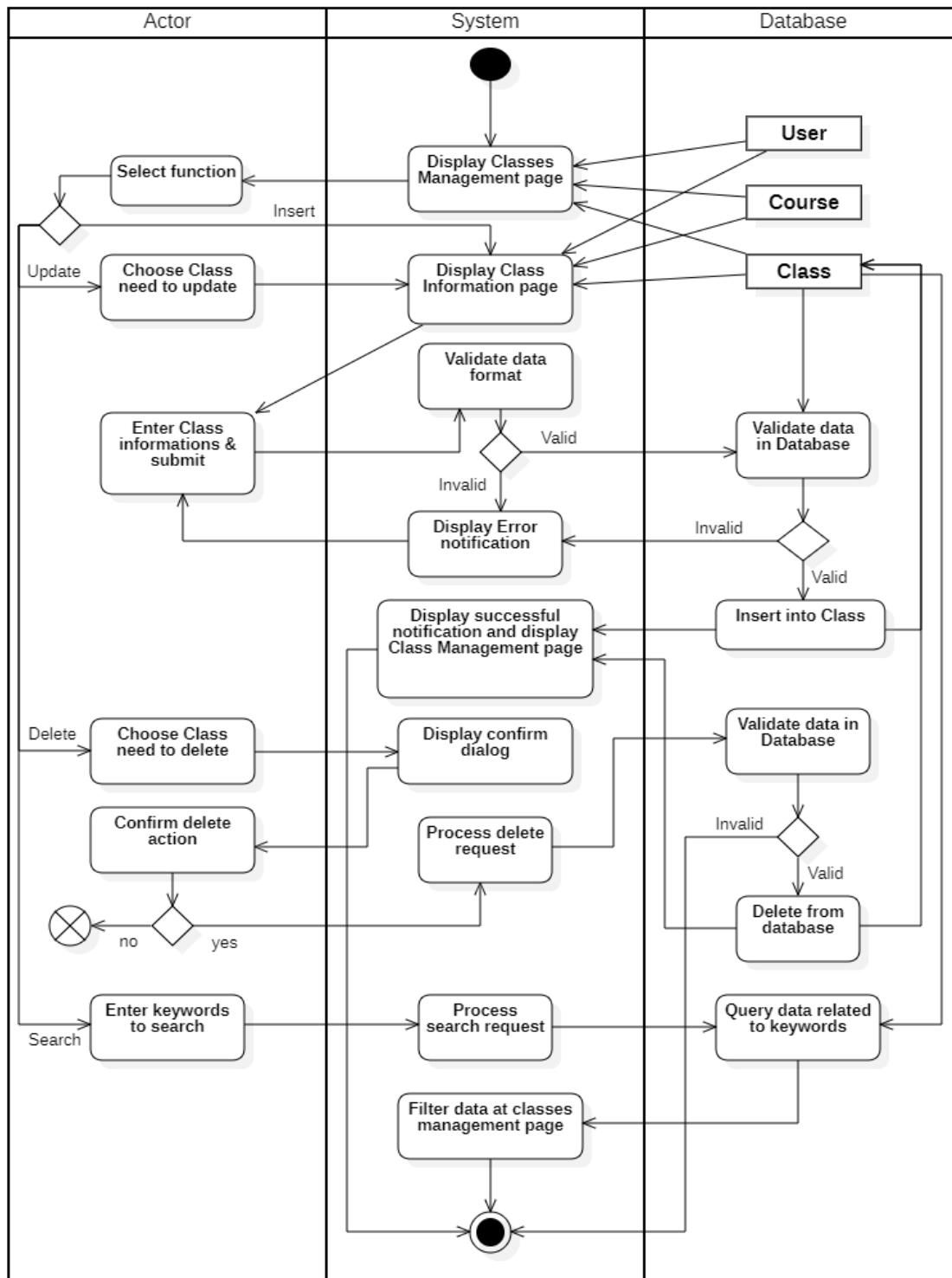


Figure 3. 8. Activity Diagram for Use case “Manage Classes”

3.2.3.1. Manage Lessons

Use-case name	Manage Lessons
Description	Manager can insert, update, delete and search Lessons information In a specific Class
Trigger	Manager select a Class in Classes management page then clicks to “Manage Lessons”
Pre-condition	<ul style="list-style-type: none"> – Manager’s status is signed in – Manager has previously accessed Classes management menu and selected a specific class.
Post-condition	<ul style="list-style-type: none"> – Lessons information successful updated in database. – Data displayed on the Lessons management page must be updated with the latest data
Basic flow	<ol style="list-style-type: none"> 1. Manager Select a Class and choose manage Lessons 2. System displays Lessons management page 3. Manager chooses one function: insert/update/delete/search Lesson information <ul style="list-style-type: none"> • Insert 4. System displays lesson information page 5. Manager enter lesson informations & submit 6. System validates lesson information in correct format 7. System validates lesson information with existed data in database 8. System stores lesson information in database 9. System display successful notification and displays Lessons management page <ul style="list-style-type: none"> • Update

	<p>10. Manager chooses one lesson to be updated</p> <p>11. System displays lesson information page</p> <p>12. Manager enter lesson informations & submit</p> <p>13. System validates lesson information in correct format</p> <p>14. System validates lesson information with existed data in database</p> <p>15. System stores lesson information in database</p> <p>16. System display successful notification and displays Lessons management page</p> <ul style="list-style-type: none"> • Delete <p>17. Choose lesson need to be delete</p> <p>18. Display confirm dialog then manager Confirm delete action</p> <p>19. If Manager confirm delete, System process delete request. Else, cancel action.</p> <p>20. System validates no external data required lesson information and selected item existed in database</p> <p>21. System delete lesson information from database</p> <p>22. System displays successful notification and display Lessons management page with the latest data.</p> <ul style="list-style-type: none"> • Search <p>23. Manager select search function and enter keywords.</p> <p>24. System processes searching data</p> <p>25. System queries data base on given keywords</p> <p>26. System filtered data at course management page (data related to keyword is retained)</p>
Alternative flow	6a. System display invalid info due to wrong format. Use-case continue at step 5.

	<p>7a. System display invalid info because data had been existed in the database. Use-case continue at step 5.</p> <p>13a. System display invalid info due to wrong format. Use-case continue at step 12.</p> <p>14a. System display invalid info because data had been existed in the database. Use-case continue at step 12.</p>
Exception flow	

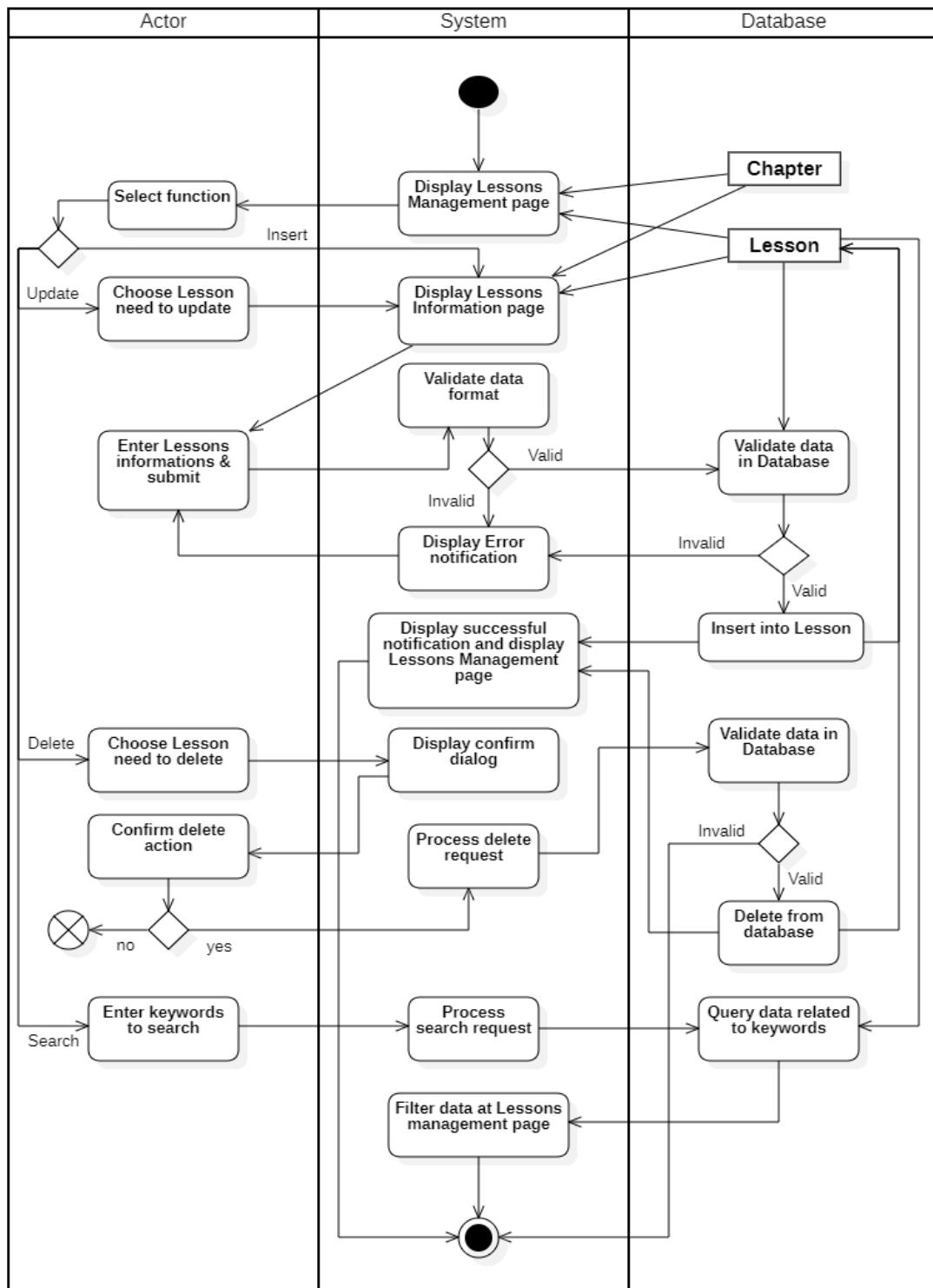
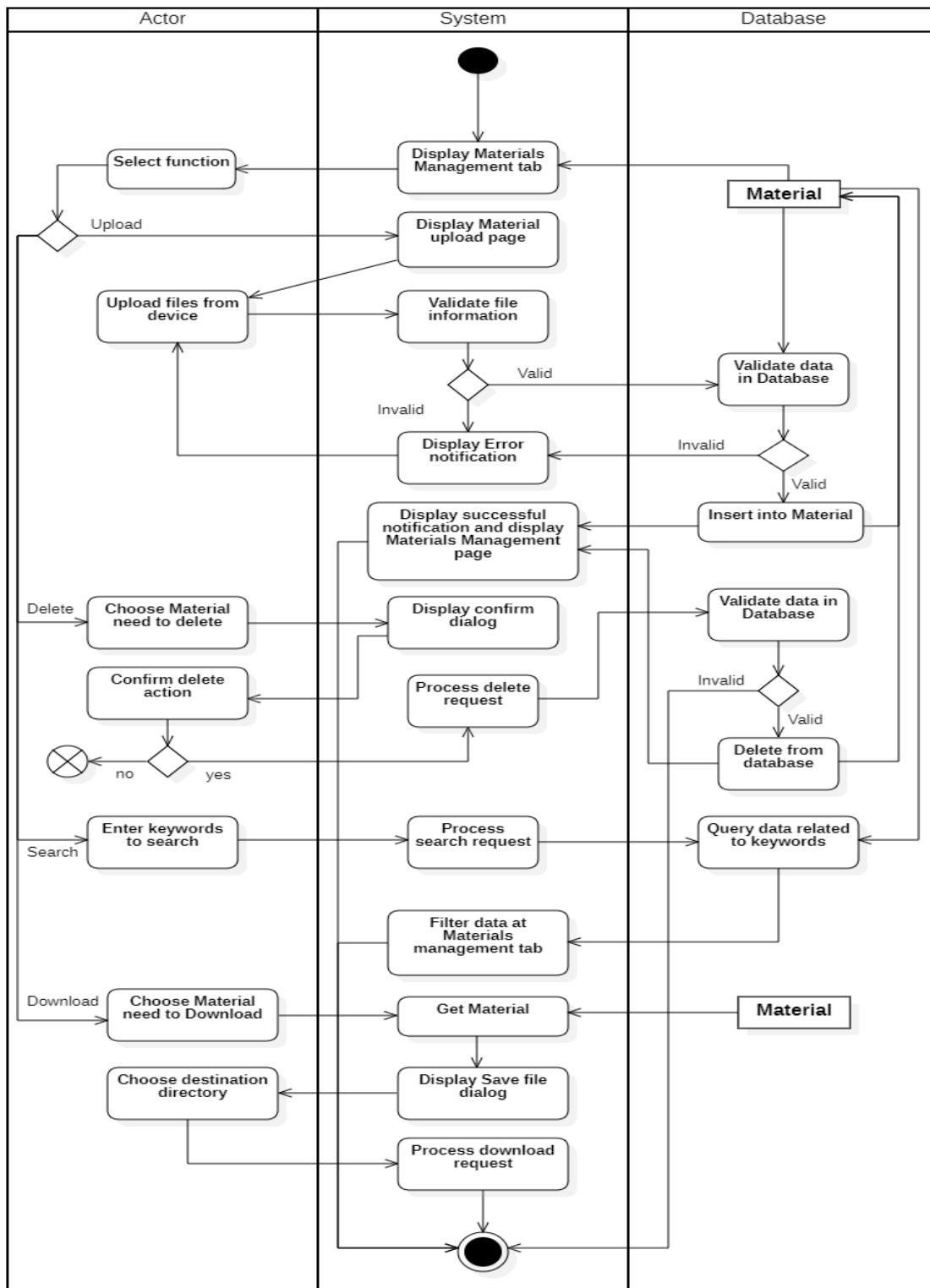


Figure 3. 9. Activity Diagram for Use case “Manage Lessons”

3.2.3.1.1. Manage Materials (For Teacher and Manager)

Use-case name	Manage Materials
Description	User can upload, download, delete and search Material information In a specific Lesson
Trigger	User select “Materials” tab after select a Lesson in Lessons Management page
Pre-condition	<ul style="list-style-type: none"> – User’s status is signed in – User has previously accessed Lessons management page and selected a specific lesson.
Post-condition	<ul style="list-style-type: none"> – Materials information successful updated in database. – Data displayed on the Materials management page must be updated with the latest data
Basic flow	<ol style="list-style-type: none"> 1. User Select a lesson and choose Materials tab. 2. System displays Materials management page 3. User chooses one function: upload/download/delete/search Material <ul style="list-style-type: none"> • Upload 4. System displays Material page 5. User upload files/materials from device 6. System validates files information in correct format 7. System validates material information with existed data in database 8. System stores material information in database 9. System display successful notification and displays Materials management page <ul style="list-style-type: none"> • Delete 10. Choose material need to be delete

	<p>11. Display confirm dialog then User Confirm delete action</p> <p>12. If User confirm delete, System process delete request. Else, cancel action.</p> <p>13. System validates no external data required material information and selected item existed in database</p> <p>14. System delete material information from database</p> <p>15. System displays successful notification and display Materials management page with the latest data.</p> <ul style="list-style-type: none"> • Search <p>16. User select search function and enter keywords.</p> <p>17. System processes searching data</p> <p>18. System queries data base on given keywords</p> <p>19. System filtered data at course management page (data related to keyword is retained)</p> <ul style="list-style-type: none"> • Download <p>20. User Choose material need to download</p> <p>21. System Get material path from database</p> <p>22. System Display Save file dialog</p> <p>23. User Choose destination directory</p> <p>24. System Process download request</p>
Alternative flow	<p>6a. System display invalid info due to wrong format. Use-case continue at step 5.</p> <p>7a. System display invalid info because data had been existed in the database. Use-case continue at step 5.</p>
Exception flow	

Figure 3. 10. Activity Diagram for Use case “**Manage Materials**”

3.2.3.1.2. Manage Attendances (For Teacher and Manager)

Use-case name	Manage Attendances
Description	User can insert, update, delete and search Attendances information in a specific Lesson
Trigger	User select “Attendances” tab after select a Lesson in Lessons Management page
Pre-condition	<ul style="list-style-type: none"> – User’s status is signed in – User has previously accessed Lessons management page and selected a specific lesson.
Post-condition	<ul style="list-style-type: none"> – Attendances information successful updated in database. – Data displayed on the Materials management page must be updated with the latest data
Basic flow	<ol style="list-style-type: none"> 1. User Select a lesson and choose Attendances tab. 2. System displays Attendances management tab with absentee table and presentee table 3. User Select a student from absentee table or presentee table <ul style="list-style-type: none"> • Mark present 4. System move selected student from absentee table to presentee table 5. System insert selected student into database <ul style="list-style-type: none"> • Mark absent 6. System move selected student from presentee table to absentee table 7. System delete selected student from database <ul style="list-style-type: none"> • Search 8. User select search function and enter keywords. 9. System processes searching data

	10. System queries data base on given keywords 11. System filtered data at course management page (data related to keyword is retained)
Alternative flow	2a. The page will initialize by default the first time you access this tab, the absent students table will be the list of students for that class and the students present table will be empty. 2b. On subsequent visits, the absent and present tables will show data edited since the last visit.
Exception flow	

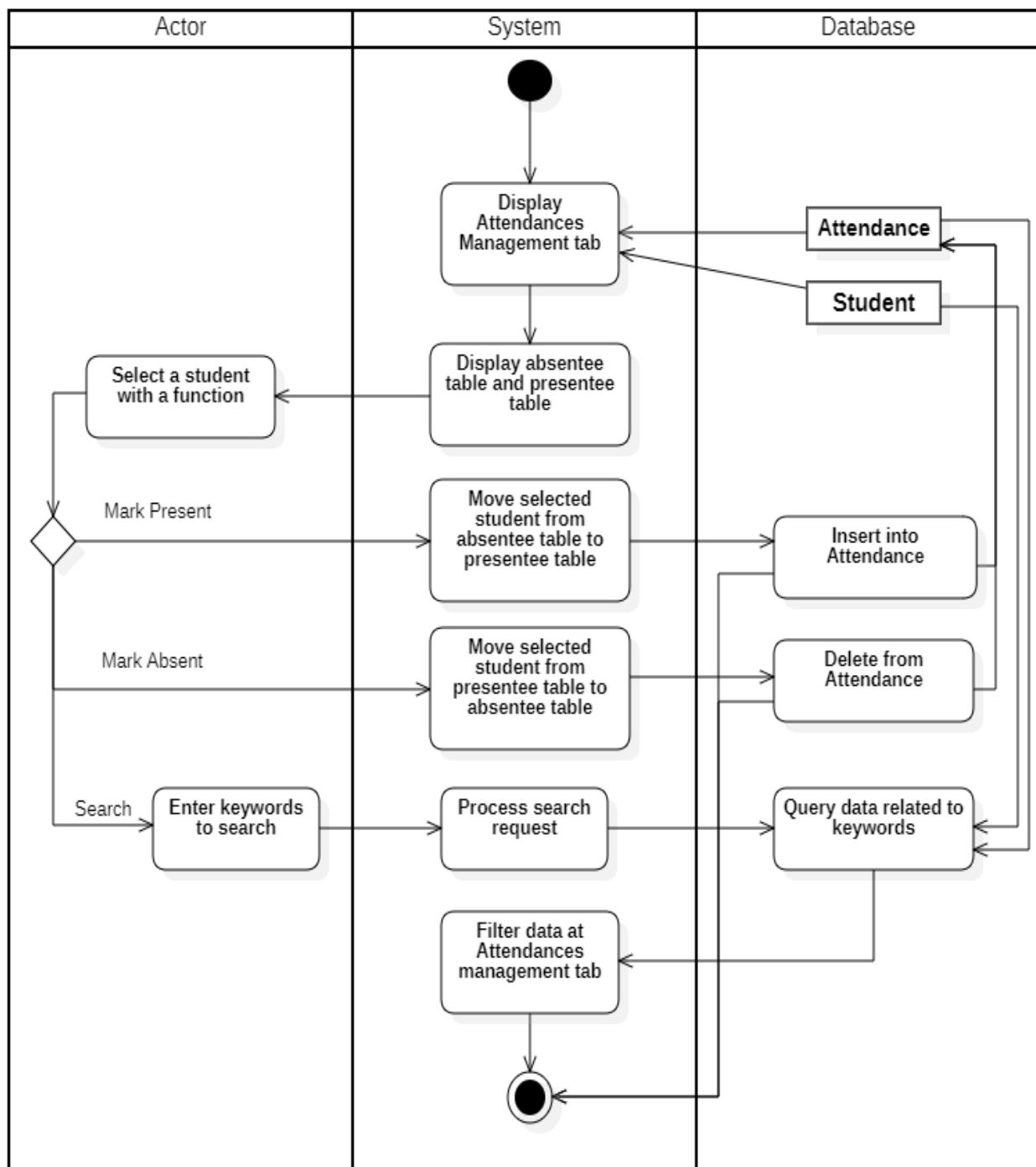


Figure 3. 11. Activity Diagram for Use case “Manage Attendances”

3.2.4. Manage Classrooms

Use-case name	Manage Classrooms
Description	Manager can insert, update, delete, search and view schedule in a specific Classrooms
Trigger	Manager clicks to “Rooms” menu at homepage
Pre-condition	Manager’s status is signed in
Post-condition	<ul style="list-style-type: none"> – Classrooms information successfully updated in database. – Data displayed on the Classrooms management page must be updated with the latest data
Basic flow	<ol style="list-style-type: none"> 1. System displays Classrooms management page 3. Manager chooses one function: insert/update/delete/search/view schedule classrooms <ul style="list-style-type: none"> • Insert 4. System displays classroom information page 5. Manager enters classroom information & submits 6. System validates classroom information in correct format 7. System validates classroom information with existing data in database 8. System stores classroom information in database 9. System displays successful notification and displays classrooms management page <ul style="list-style-type: none"> • Update 10. Manager chooses one classroom to be updated 11. System displays classroom information page 12. Manager enters classroom information & submits 13. System validates classroom information in correct format

	<ol style="list-style-type: none">14. System validates classroom information with existed data in database15. System stores classroom information in database16. System display successful notification and displays classrooms management page• Delete17. Choose classroom need to be delete18. Display confirm dialog then manager Confirm delete action19. If Manager confirm delete, System process delete request. Else, cancel action.20. System validates no external data required classroom information and selected item existed in database21. System delete classroom information from database22. System displays successful notification and display classrooms management page with the latest data.• Search23. Manager select search function and enter keywords.24. System processes searching data25. System queries data base on given keywords26. System filtered data at classroom management page (data related to keyword is retained)• View Schedule27. Manager choose a classroom want to view schedule28. System get data from database and generate a schedule for selected room29. System display schedule as timetable with class information
--	---

Alternative flow	6a. System display invalid info due to wrong format. Use-case continue at step 5. 7a. System display invalid info because invalid in database. Use-case continue at step 5. 13a. System display invalid info due to wrong format. Use-case continue at step 12. 14a. System display invalid info because data had been existed in the database. Use-case continue at step 12.
Exception flow	

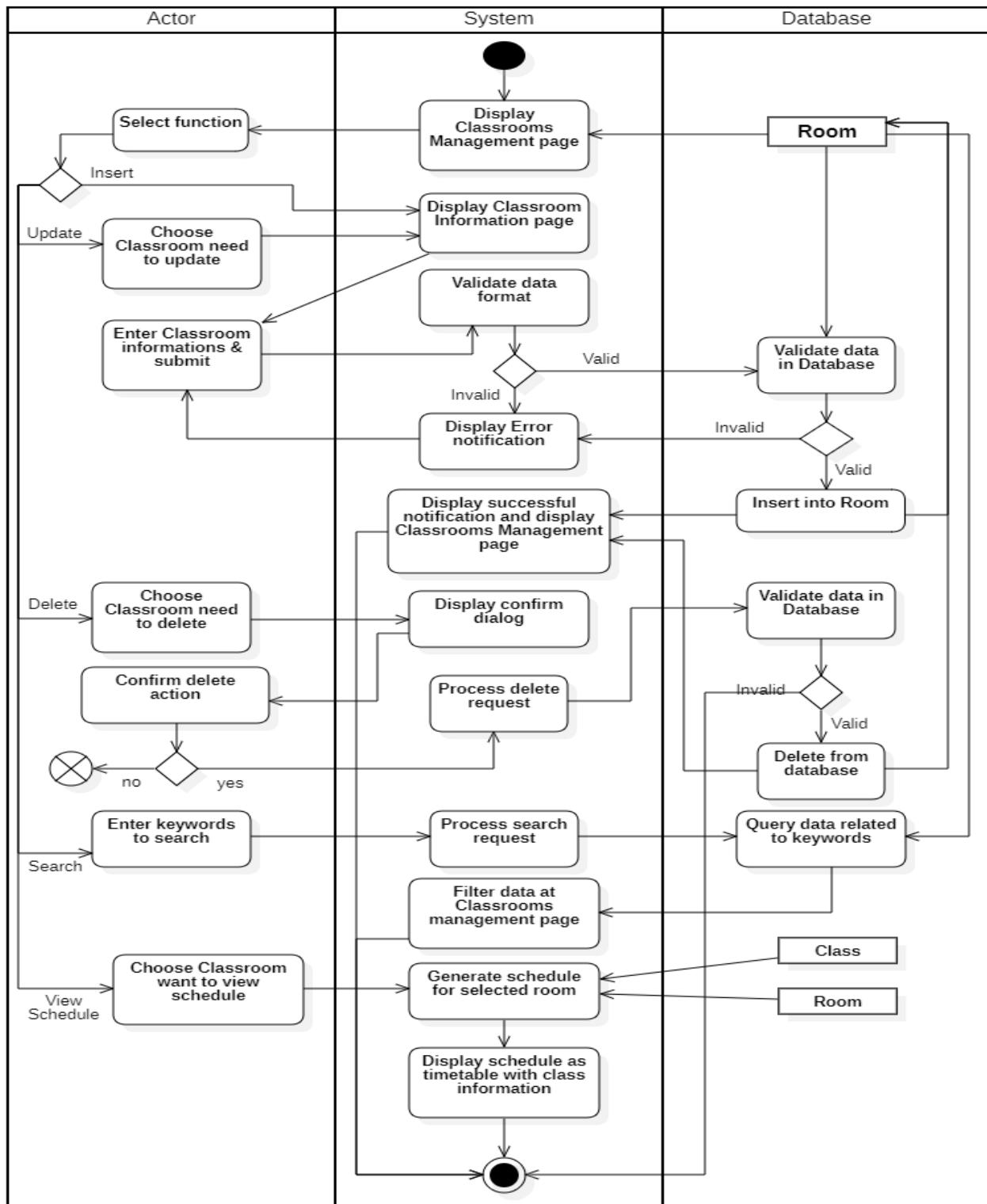


Figure 3. 12. Activity Diagram for Use case “Manage Classrooms”

3.2.5. Manage Users

Use-case name	Manage Users
Description	Manager can insert, update, delete, search Users and Enroll user in a Class
Trigger	Manager clicks to “Users” menu at homepage
Pre-condition	Manager’s status is signed in
Post-condition	<ul style="list-style-type: none"> – Course’s information successfully updated in database. – Data displayed on the Course management page must be updated with the latest data.
Basic flow	<ol style="list-style-type: none"> 1. System displays Users management page 2. Manager chooses one function: insert/update/delete/search users and enroll in a class <ul style="list-style-type: none"> • Insert 3. System displays user information page 4. Manager enter user informations & submit 5. System validates user information in correct format 6. System validates user information with existed data in database 7. System stores user information in database 8. System displays successful notification and displays users management page <ul style="list-style-type: none"> • Update 9. Manager chooses one user to be updated 10. System displays user information page 11. Manager enter user informations & submit 12. System validates user information in correct format

	<ol style="list-style-type: none">13. System validates user information with existed data in database14. System stores user information in database15. System display successful notification and displays users management page<ul style="list-style-type: none">• Delete16. Choose user need to be delete17. Display confirm dialog then manager Confirm delete action18. If Manager confirm delete, System process delete request. Else, cancel action.19. System validates no external data required user information and selected item existed in database20. System delete user information from database21. System displays successful notification and display users management page with the latest data.• Search22. Manager select search function and enter keywords.23. System processes searching data24. System queries data base on given keywords25. System filtered data at user management page (data related to keyword is retained)• Enroll in a Class26. Manager select a Student to enroll in a Class27. System display of available classes28. Manager choose a Class in the list29. System display confirmation dialog30. If Manager confirm enrollment, System process request. Else, cancel
--	---

	<p>action.</p> <p>31. System validate the enrollment in database</p> <p>32. System displays successful notification</p>
Alternative flow	<p>5a. System display invalid info due to wrong format. Use-case continue at step 4.</p> <p>6a. System display invalid info because data had been existed in the database. Use-case continue at step 4.</p> <p>12a. System display invalid info due to wrong format. Use-case continue at step 11.</p> <p>13a. System display invalid info because data had been existed in the database. Use-case continue at step 11.</p> <p>31a. System display error notification because invalid in database. Use case continue at step 28.</p>
Exception flow	

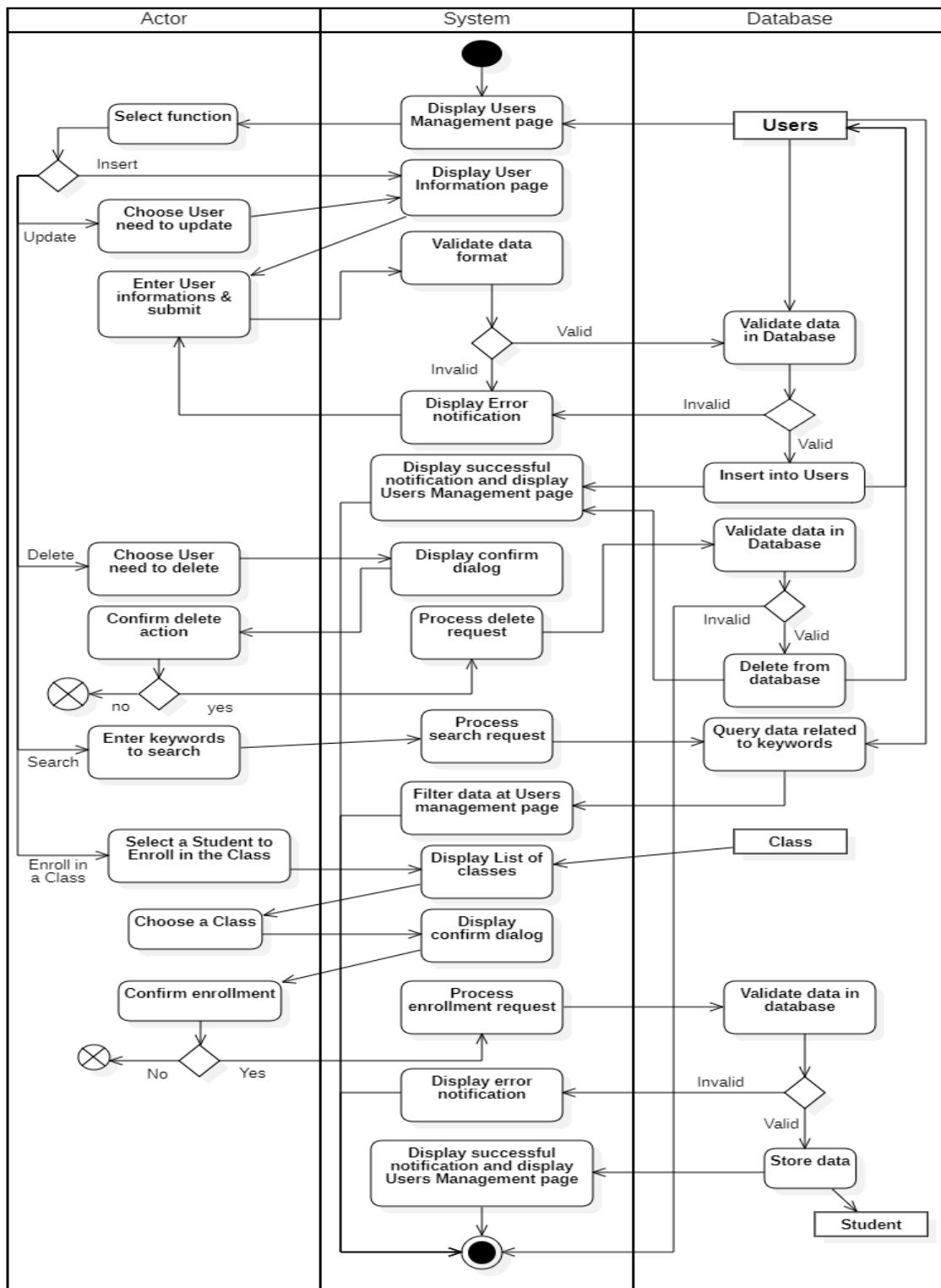


Figure 3. 13. Activity Diagram for Use case “Manage Users”

3.2.6. Export Class report

Use-case name	Export class report
Description	Manager can export class report about class attendances
Trigger	Manager clicks to “Export report” button at Classes Management Page
Pre-condition	Manager’s status is signed in Manager was entered Classes Management Page
Post-condition	<ul style="list-style-type: none"> • Class report export at destination directory successful • Class report contains suitable information
Basic flow	<ol style="list-style-type: none"> 1. System display list of Classes 2. Manager choose a class in the list 3. System generate Class report for selected Class 4. System display Save file dialog for manager 5. Manager choose file destination directory for the report 6. System export the report to PDF into destination directory 7. System display successful notification
Alternative flow	
Exception flow	

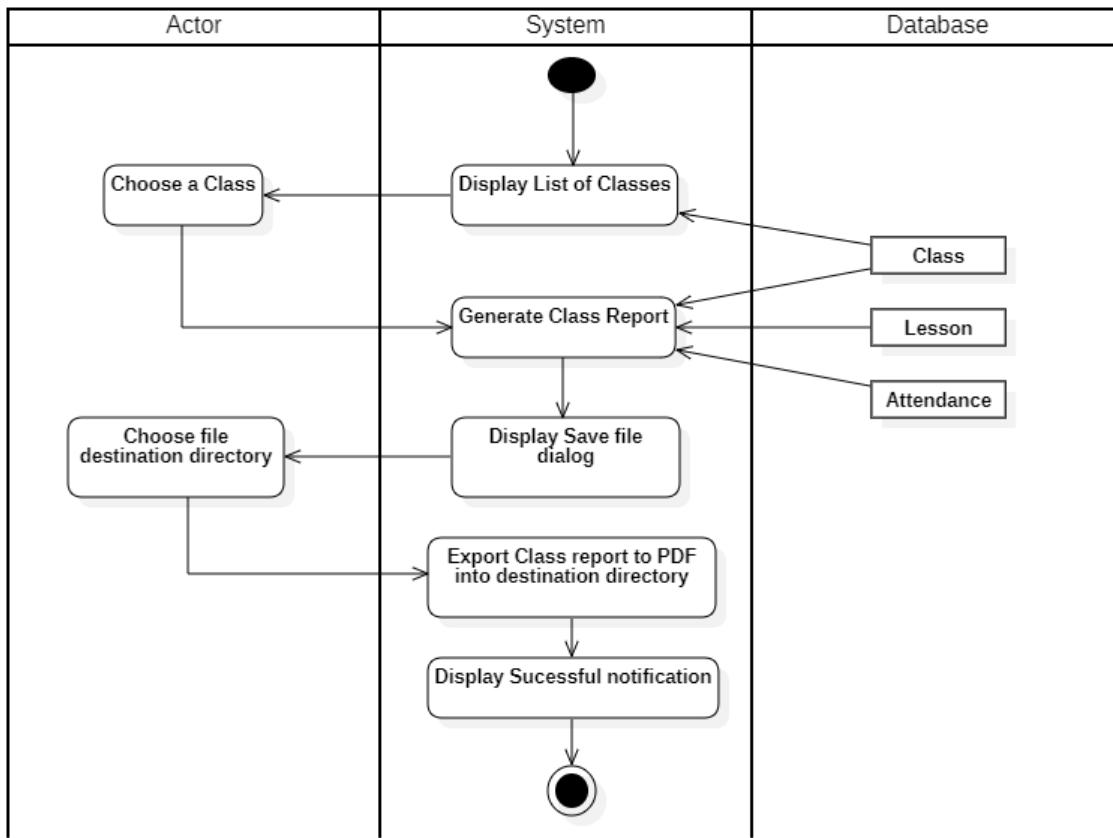


Figure 3. 14. Activity Diagram for Use case “Export Class Report”

3.2.7. Analyze

Use-case name	Analyze
Description	Manager can analyze the performance of courses and classes through visual chart
Trigger	Manager clicks to “General” menu at homepage
Pre-condition	Manager’s status is signed in
Post-condition	Data & Chart display on General page with data queried from database
Basic flow	<ol style="list-style-type: none"> 1. System display dashboard page with some basic metric 2. Manager select a analysis category

	<ul style="list-style-type: none"> • <i>Number of students through lessons in a Class</i> <p>3. System display list of classes 4. Manager choose a class want to analyze 5. System query data from database 6. System generate and display visual chart base on data queried</p> <ul style="list-style-type: none"> • <i>Number of classes of each course</i> <p>7. System query data from database 8. System generate and display visual chart base on data queried</p>
Alternative flow	
Exception flow	

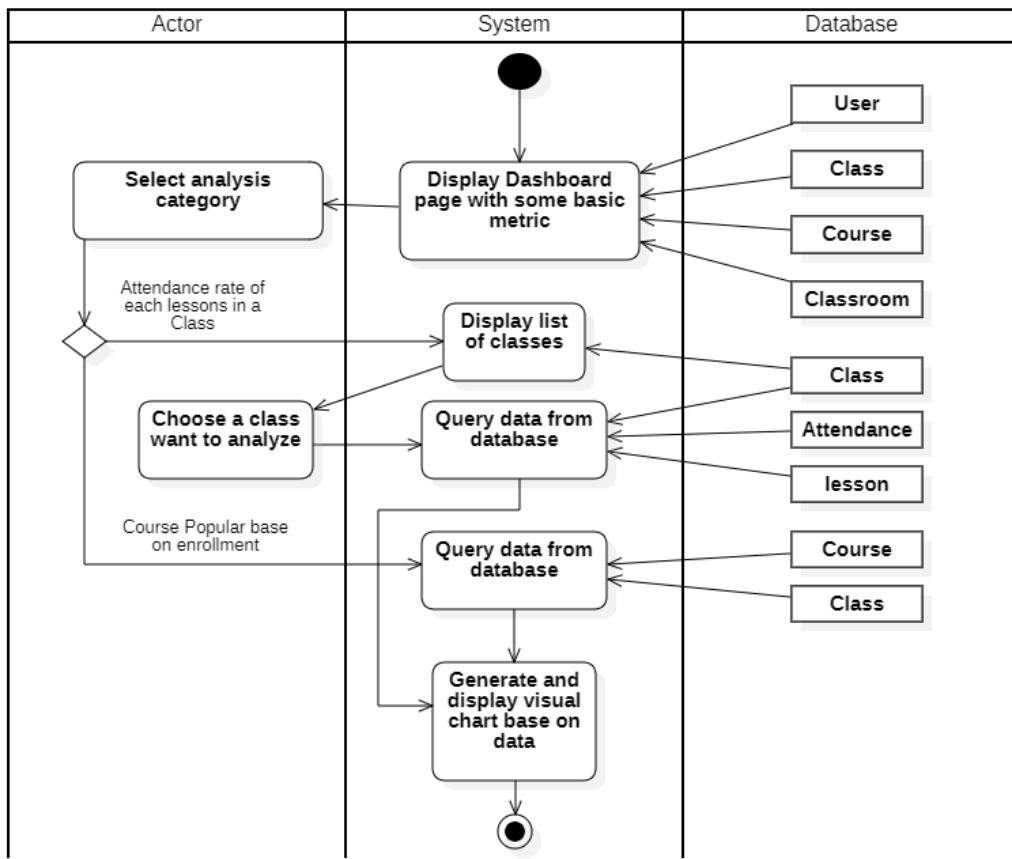


Figure 3. 15. Activity Diagram for Use case "Analyze"

3.2.8. Logout (For all users)

Use-case name	Logout
Description	User log out from system
Trigger	User clicks to “Sign out” button at homepage
Pre-condition	User’s status is signed in
Post-condition	<ul style="list-style-type: none"> – User logged out from system – System display Login page
Basic flow	<ol style="list-style-type: none"> 1. System display Confirm dialog 2. Manager confirm action log out

	3. Display Login page
Alternative flow	2a. System cancel action due to Manager no confirm action
Exception flow	

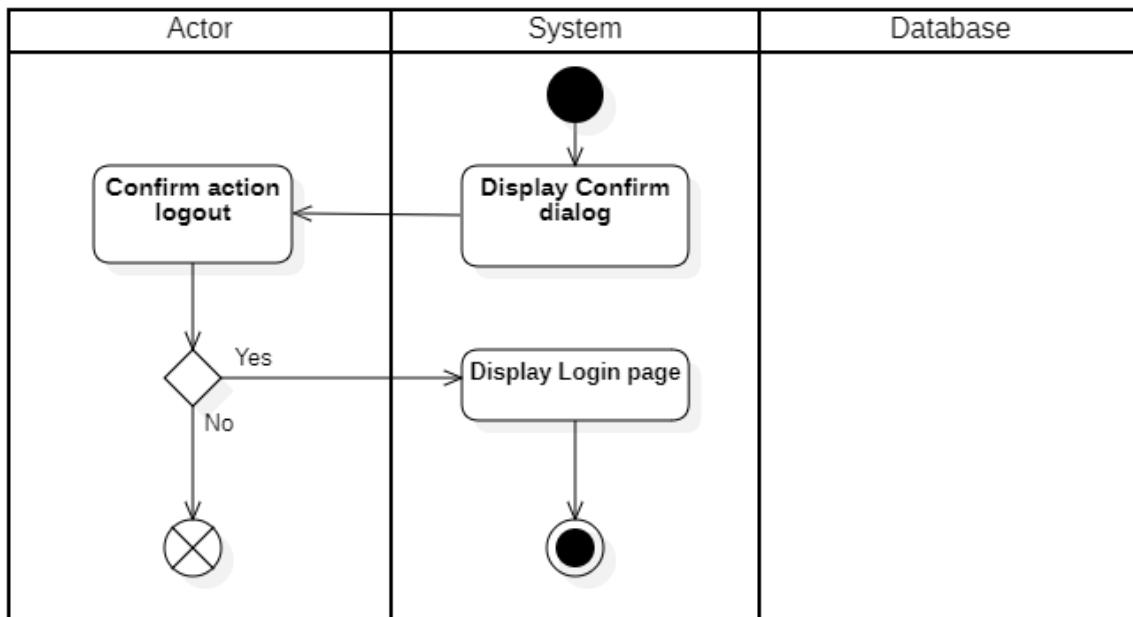


Figure 3. 16. Activity Diagram for Use case “Logout”

3.2.9. Explore Courses (For User)

Use-case name	Explore Courses
Description	Explore exists Courses in the center
Trigger	User clicks to “Courses” menu at homepage
Pre-condition	User’s status is signed in
Post-condition	Courses menu display Courses queried from database
Basic flow	1. System display List of Courses queried from database 2. Manager select search function

	3. Manager enter keywords to search 4. System process search request 5. System query data from database base on given keywords 6. System filter data at Courses page
Alternative flow	
Exception flow	

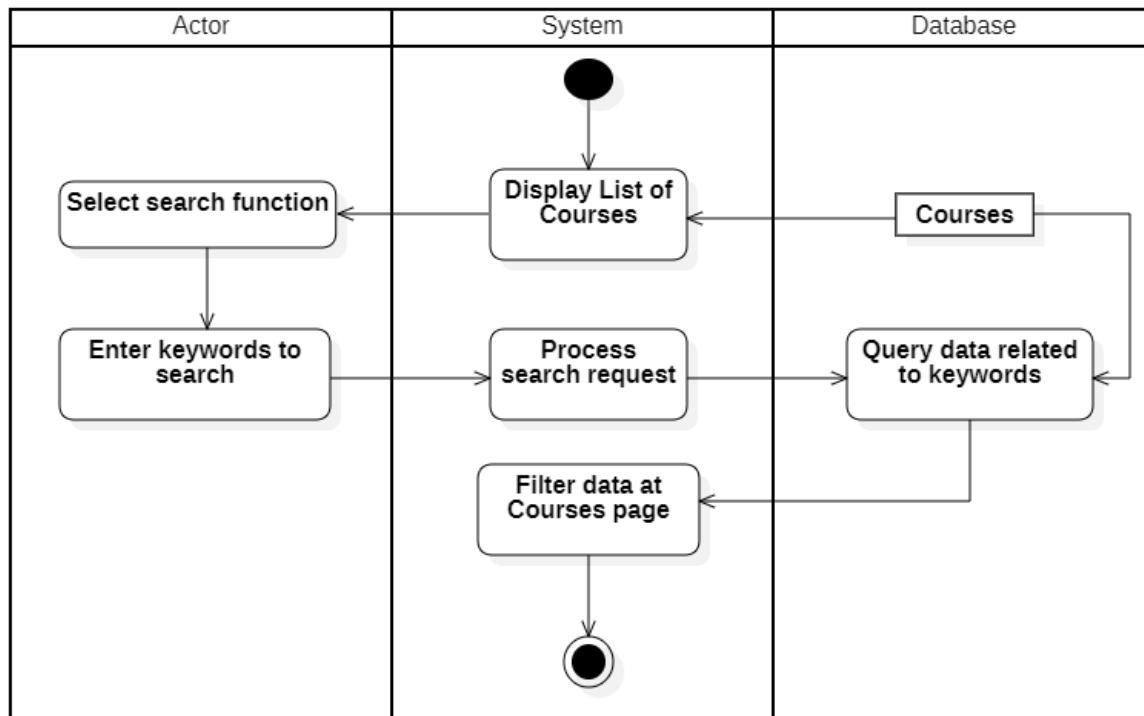


Figure 3. 17. Activity Diagram for Use case “Explore Courses”

3.2.9.1. Explore Chapters (For User)

Use-case name	Explore Chapters
Description	Explore exists Chapters in a specific Course
Trigger	User clicks to “Chapters” after select a Course at courses page
Pre-condition	User’s status is signed in User already access Courses page
Post-condition	Chapters menu display Chapter in a Course queried from database
Basic flow	<ol style="list-style-type: none"> 1. System display List of Chapters in a Course queried from database 2. Manager select search function 3. Manager enter keywords to search 4. System process search request 5. System query data from database base on given keywords 6. System filter data at Chapters page
Alternative flow	
Exception flow	

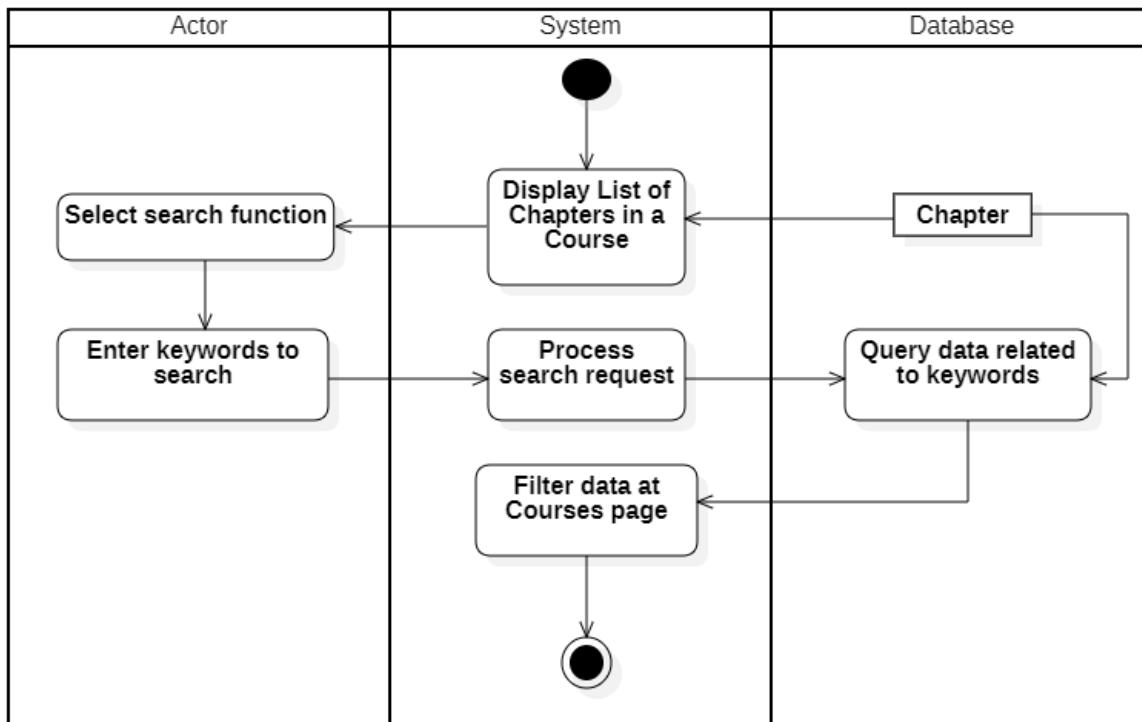


Figure 3. 18. Activity Diagram for Use case “Explore Chapters”

3.2.10. View Schedule (For User)

Use-case name	View Schedule
Description	View Studying/Teaching Schedule in English Center
Trigger	User clicks to “Schedule” menu at homepage
Pre-condition	User’s status is signed in
Post-condition	Menu generates a clearly schedule for User
Basic flow	<ol style="list-style-type: none"> 1. System Query data from database 2. Generate schedule for User 3. System Display schedule as timetable with class information
Alternative	

flow	
Exception flow	

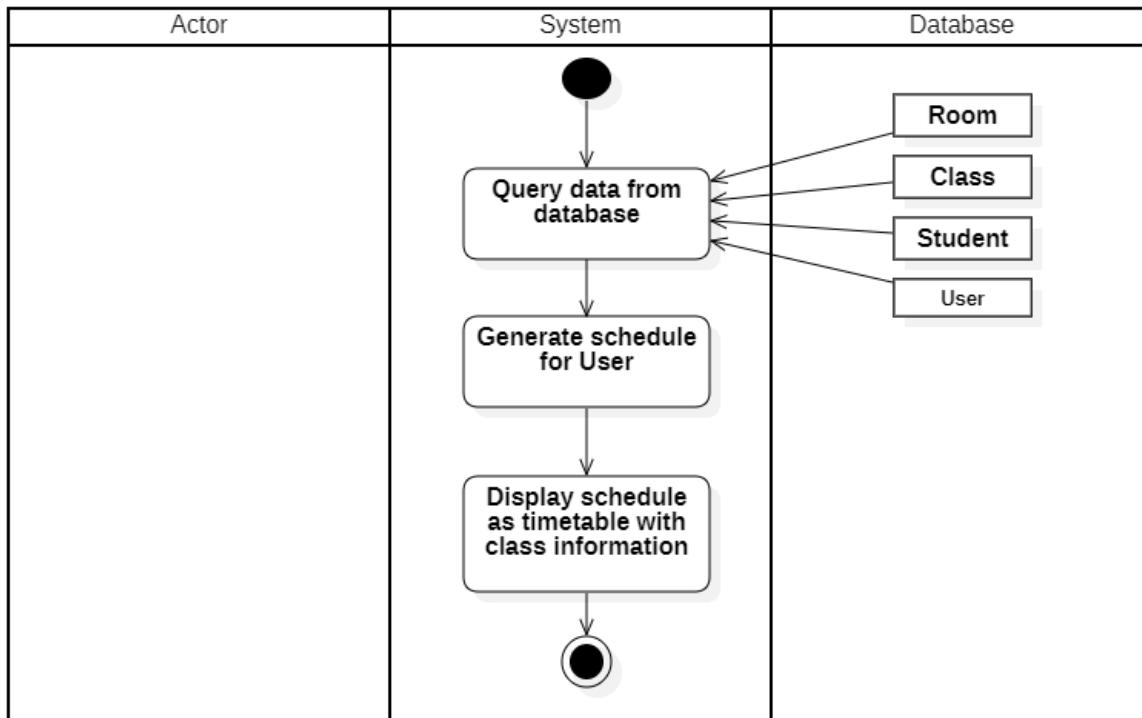


Figure 3. 19. Activity Diagram for Use case “View Schedule”

3.2.11. Manage Classes (For Teacher)

Use-case name	Manage Classes
Description	Manage Classes that Teacher is currently participating in
Trigger	Teacher clicks to “Classes” menu at homepage
Pre-condition	Teacher’s status is signed in
Post-condition	Classes menu display list of Classes that teacher has participated
Basic flow	1. System display List of Classes queried from database

	2. Teacher select search function 3. Teacher enter keywords to search 4. System process search request 5. System query data from database base on given keywords 6. System filter data at Classes page
Alternative flow	
Exception flow	

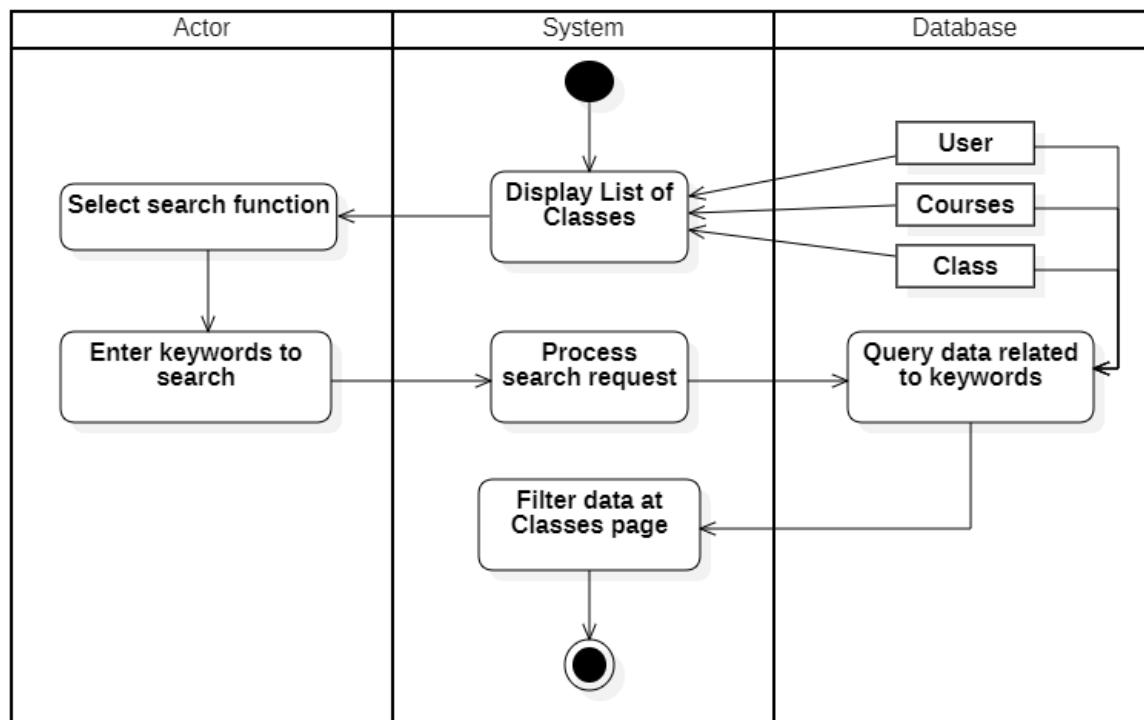


Figure 3. 20. Activity Diagram for Use case “Manage Classes”

3.2.11.1. Manage Lessons (For Teacher)

Use-case name	Manage Lessons
Description	Manage exists Lessons in a specific Class
Trigger	Teacher clicks to “View Lessons” after select a Class at courses page
Pre-condition	Teacher’s status is signed in Teacher already access Classes page
Post-condition	Lessons menu display Lessons in a Class queried from database
Basic flow	<ol style="list-style-type: none"> 1. System display List of Lessons in a Class queried from database 2. Teacher select search function 3. Teacher enter keywords to search 4. System process search request 5. System query data from database base on given keywords 6. System filter data at Lessons page
Alternative flow	
Exception flow	

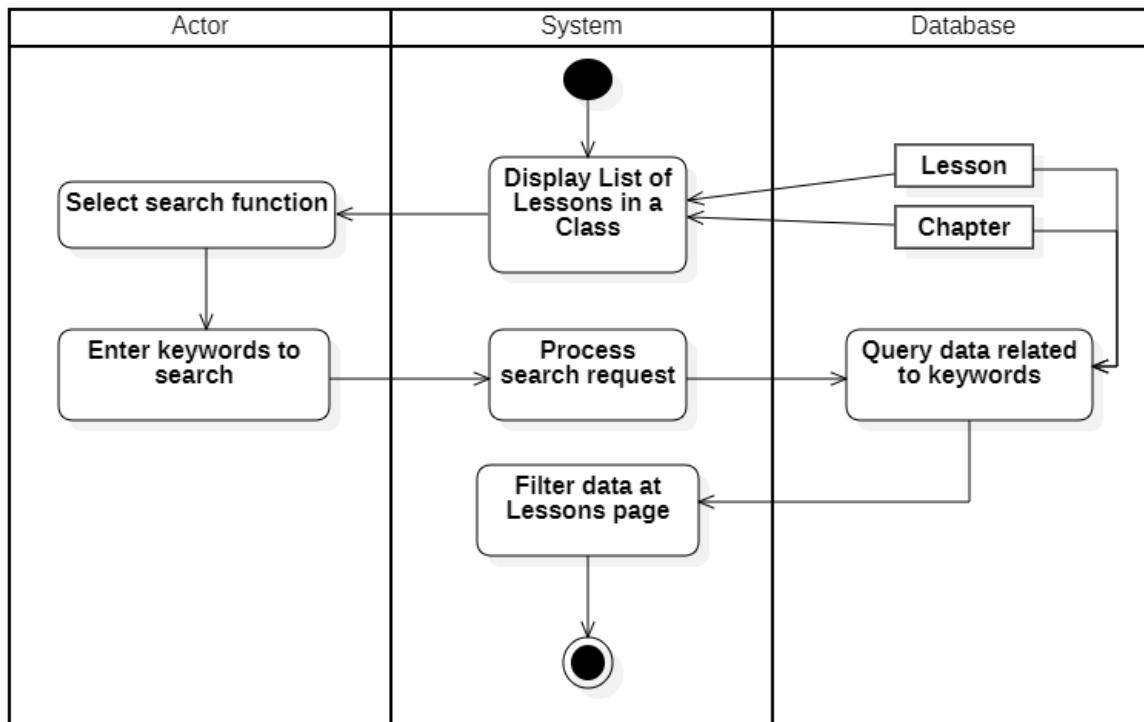


Figure 3. 21Activity Diagram for Use case “Manage Lessons”

3.2.12. View Classrooms

Use-case name	View Classrooms
Description	Teacher can search and view schedule in a specific Classrooms
Trigger	Teacher clicks to “Rooms” menu at homepage
Pre-condition	Teacher’s status is signed in
Post-condition	Data displayed on the Classrooms management page must be updated with the latest data
Basic flow	1. System displays Classrooms management page

	<p>2. Teacher chooses one function: search/view schedule classrooms</p> <ul style="list-style-type: none"> • Search <p>3. Teacher select search function and enter keywords.</p> <p>4. System processes searching data</p> <p>5. System queries data base on given keywords</p> <p>6. System filtered data at classrooms page (data related to keyword is retained)</p> <ul style="list-style-type: none"> • View Schedule <p>7. Teacher choose a classroom want to view schedule</p> <p>8. System get data from database and generate a schedule for selected room</p> <p>9. System display schedule as timetable with class information</p>
Alternative flow	
Exception flow	

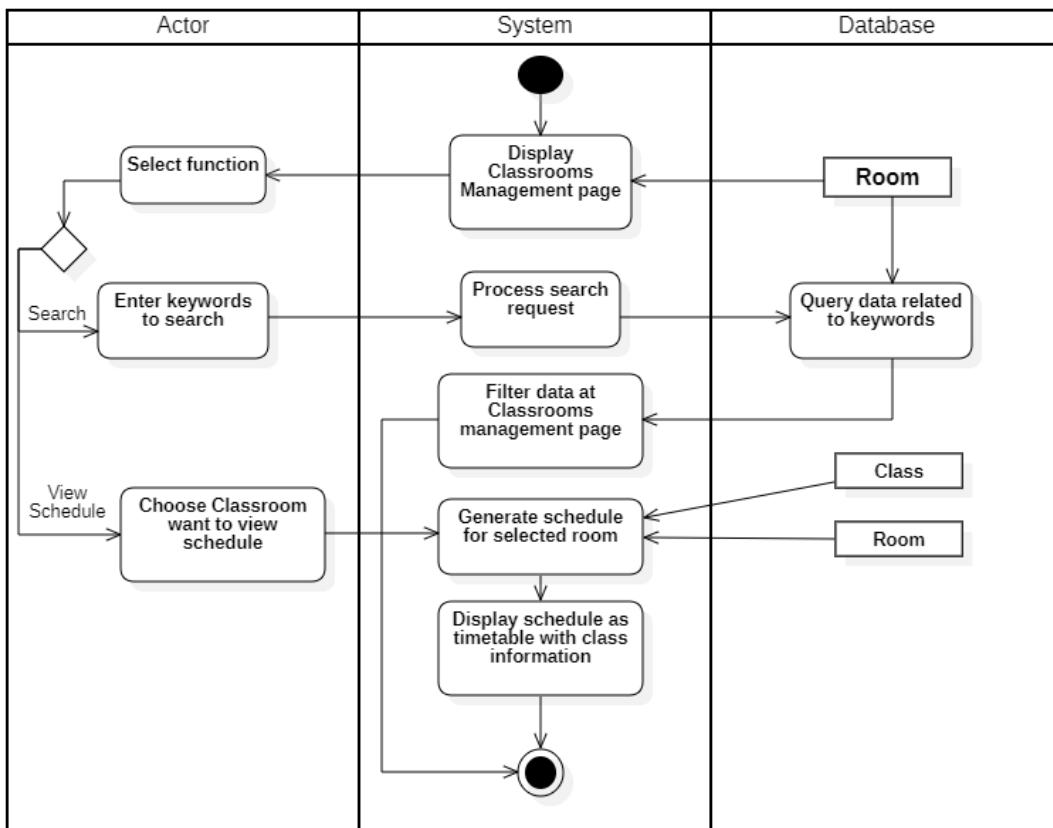


Figure 3. 22. Activity Diagram for Use case “View Classrooms”

3.2.13. View Classes

Use-case name	View Classes
Description	View Classes that Student is currently enrolled
Trigger	Student clicks to “Classes” menu at homepage
Pre-condition	Student’s status is signed in
Post-condition	Classes menu display list of Classes that Student has enrolled
Basic flow	<ol style="list-style-type: none"> 1. System display List of Classes queried from database 2. Student select search function 3. Student enter keywords to search 4. System process search request

	5. System query data from database base on given keywords 6. System filter data at Classes page
Alternative flow	
Exception flow	

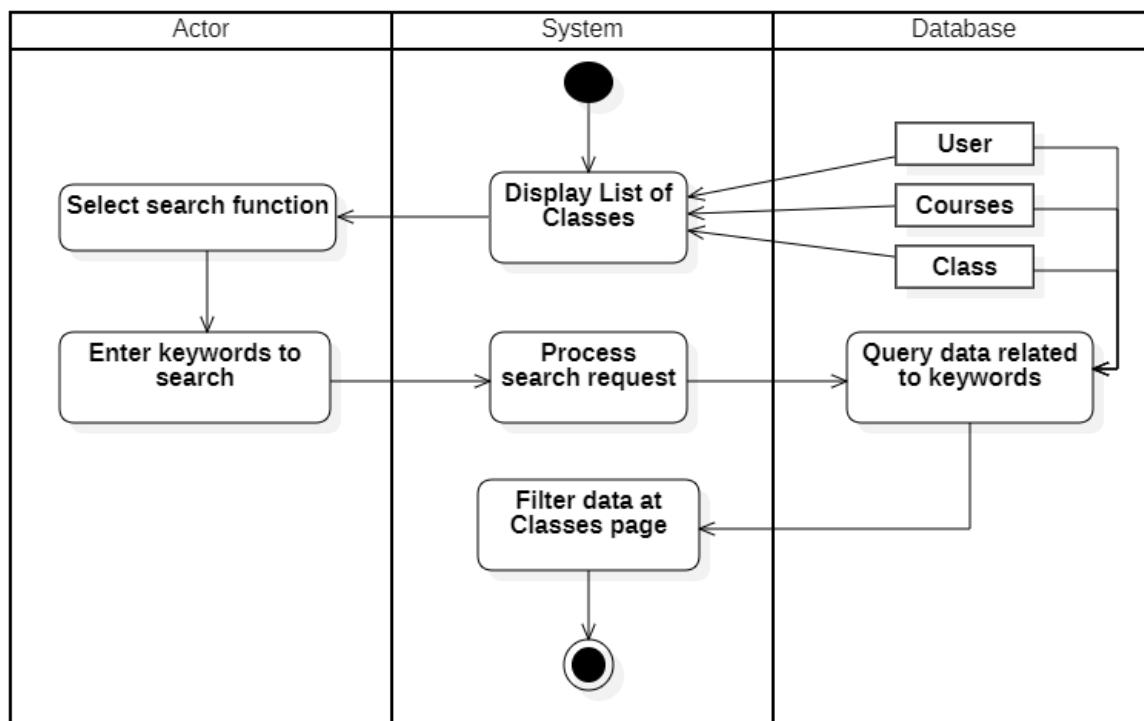


Figure 3. 23. Activity Diagram for Use case “View Classes”

3.2.13.1. View Lessons

Use-case name	View Lessons
Description	View exists Lessons in a specific Class
Trigger	Student clicks to “View Lessons” after select a Class at courses page
Pre-condition	Student’s status is signed in Student already access Classes page
Post-condition	Lessons menu display Lessons in a Class queried from database
Basic flow	<ol style="list-style-type: none"> 1. System display List of Lessons in a Class queried from database 2. Student select search function 3. Student enter keywords to search 4. System process search request 5. System query data from database base on given keywords 6. System filter data at Lessons page
Alternative flow	
Exception flow	

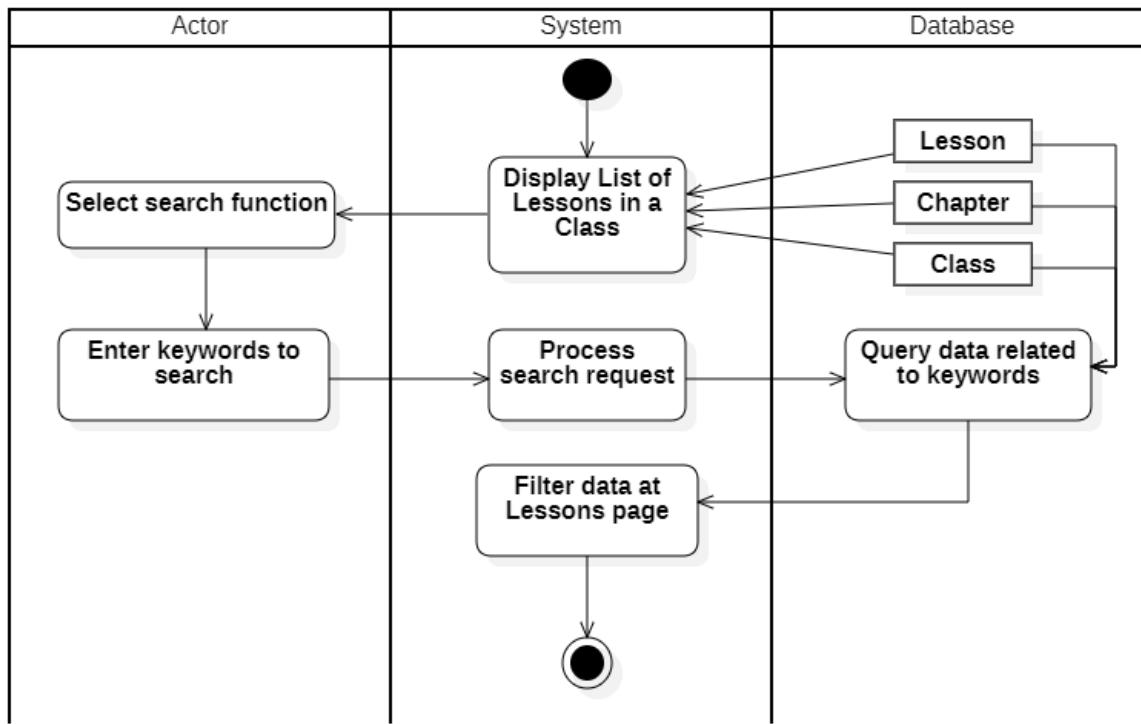


Figure 3. 24. Activity Diagram for Use case “View Lessons”

3.2.13.1.1. View Materials

Use-case name	View Materials
Description	Student can search and download Material information In a specific Lesson
Trigger	Student select “Materials” tab after select a Lesson in Lessons page
Pre-condition	<ul style="list-style-type: none"> – Student’s status is signed in – Student has previously accessed Lessons management page and selected a specific lesson.
Post-condition	Data displayed on the Materials page must be updated with the latest data
Basic flow	<ol style="list-style-type: none"> 1. Student Select a lesson and choose Materials tab. 2. System displays Materials management page 3. Student chooses one function: download/search Material <ul style="list-style-type: none"> • Search

	4. Student select search function and enter keywords. 5. System processes searching data 6. System queries data base on given keywords 7. System filtered data at Materials management page (data related to keyword is retained) <ul style="list-style-type: none"> • Download 8. Student Choose material need to download 9. System Get material path from database 10. System Display Save file dialog 11. Student Choose destination directory 12. System Process download request
Alternative flow	
Exception flow	

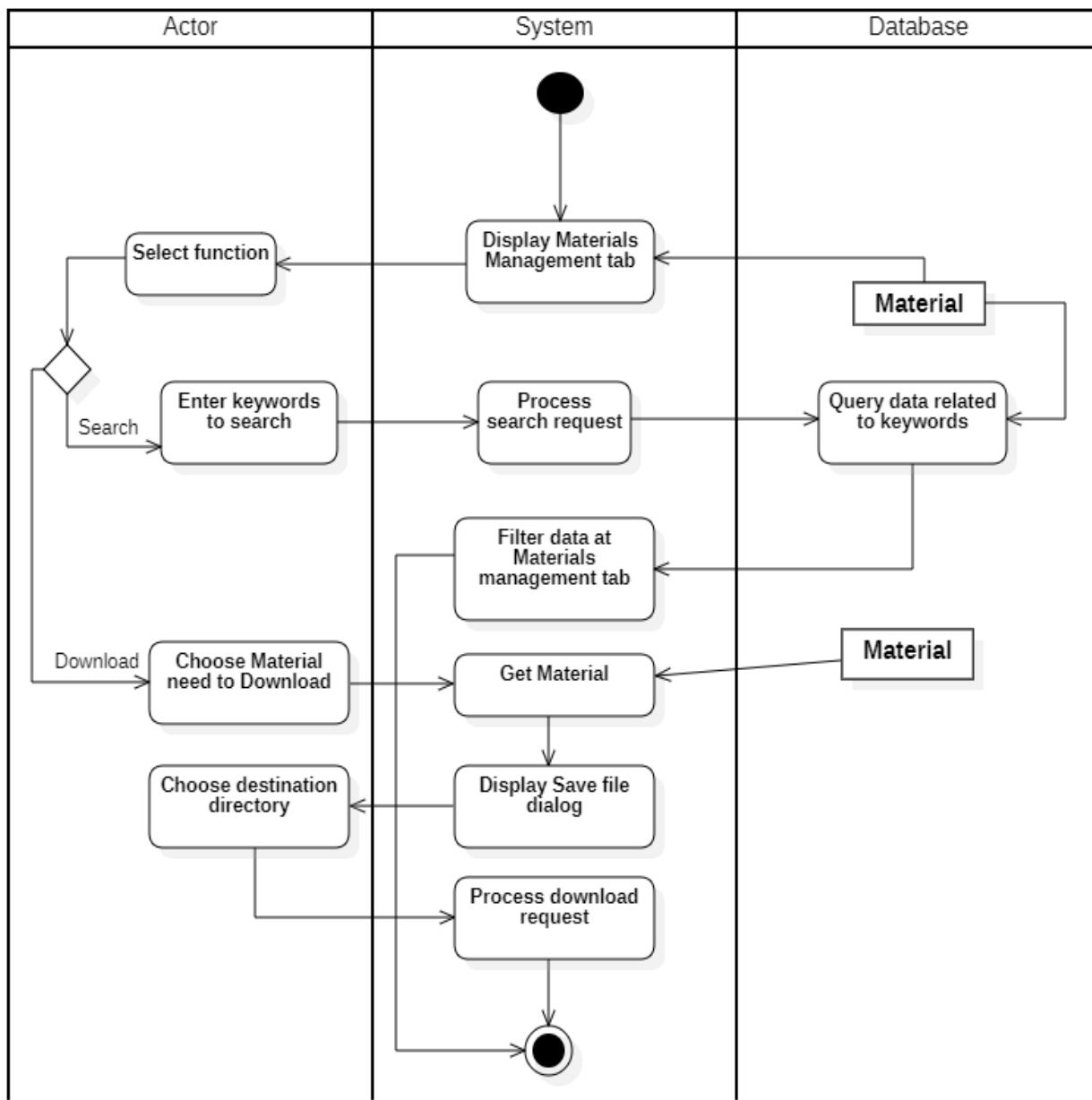


Figure 3. 25. Activity Diagram for Use case “View Materials”

3.3. Sequence Diagram and Class Diagram

3.3.1. Login

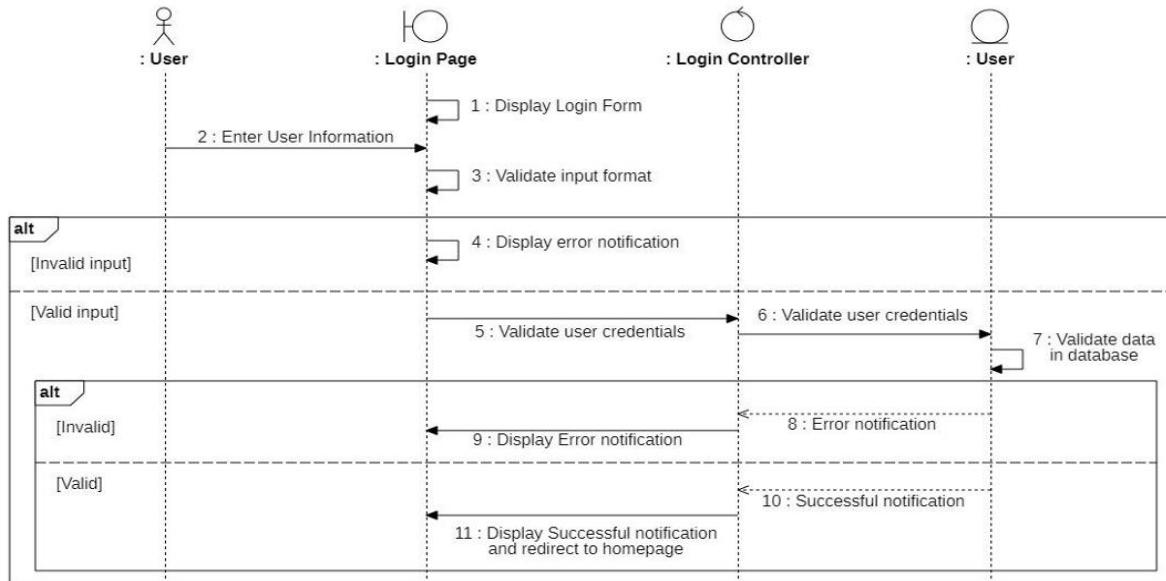


Figure 3. 26. Sequence Diagram for "Login"

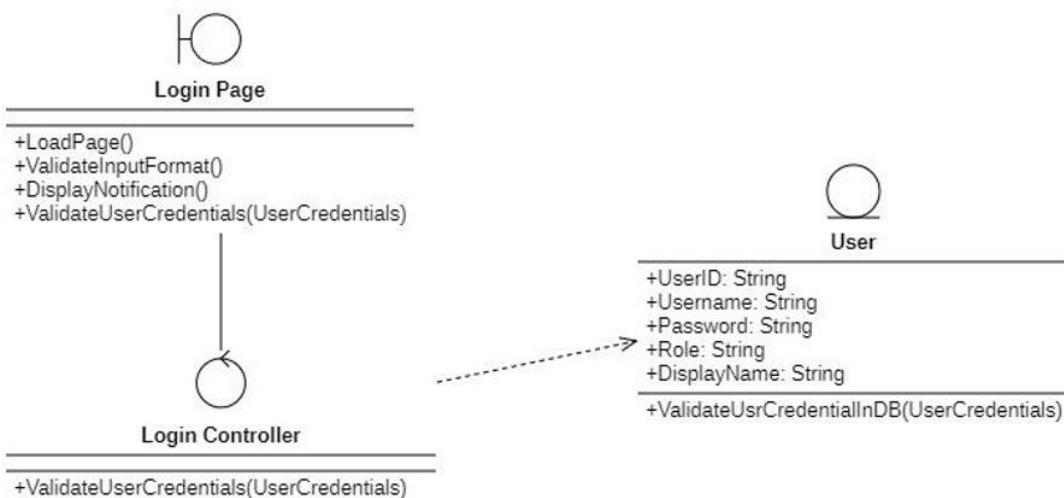


Figure 3. 27. Class Diagram for "Login"

3.3.2. Manage Courses

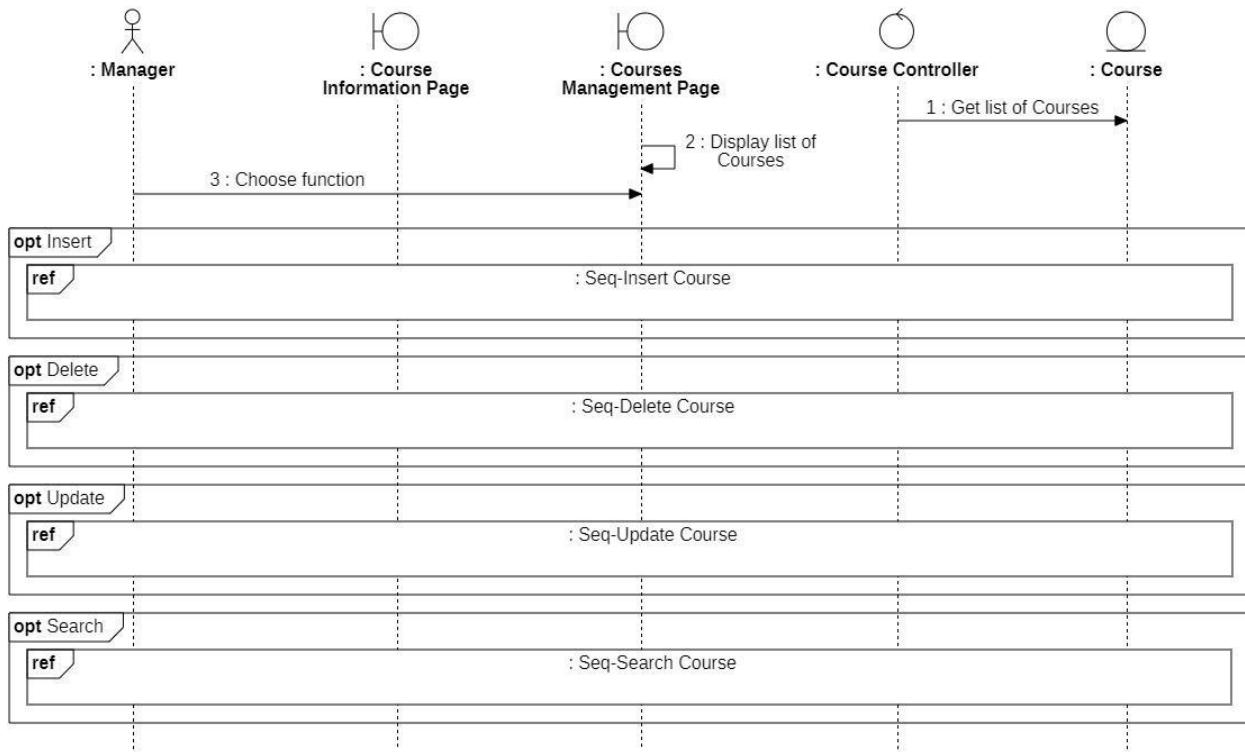


Figure 3. 28.Sequence Diagram for “Manage Courses”

– Insert

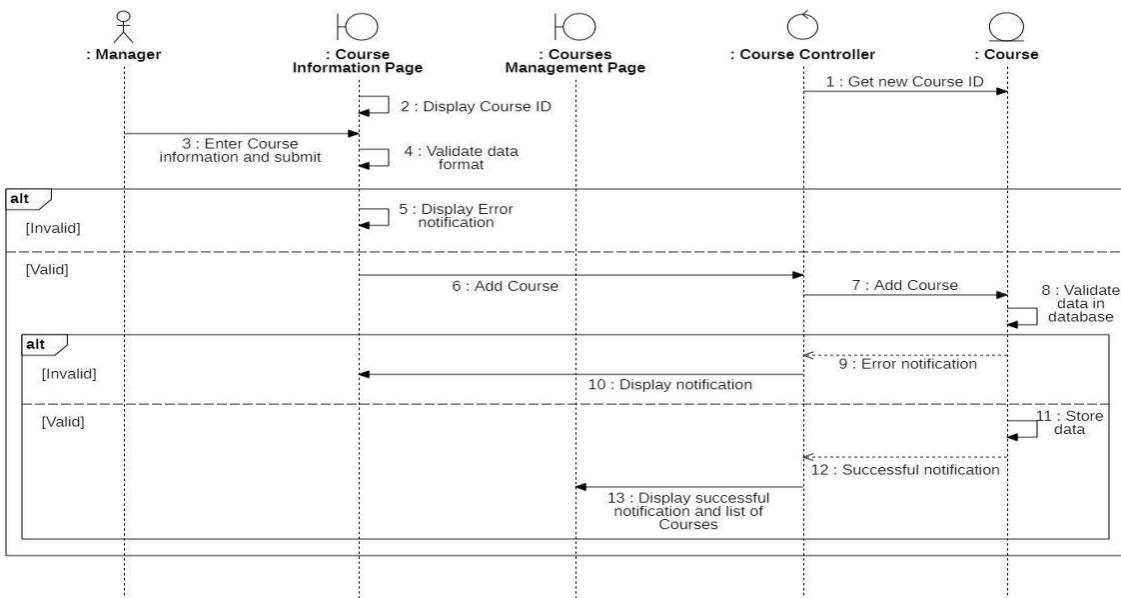


Figure 3. 29. Sequence Diagram for “Insert Course”

– Update

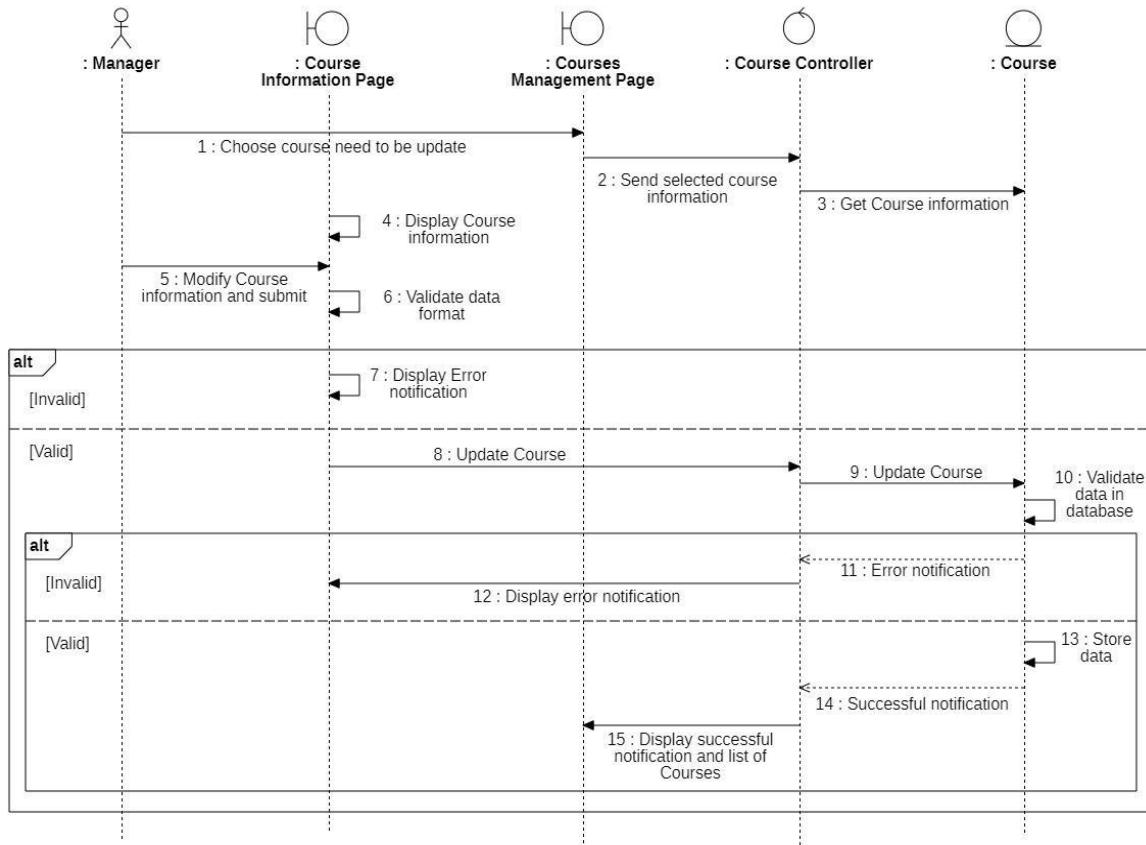


Figure 3. 30. Sequence Diagram for “Update Course”

– Delete

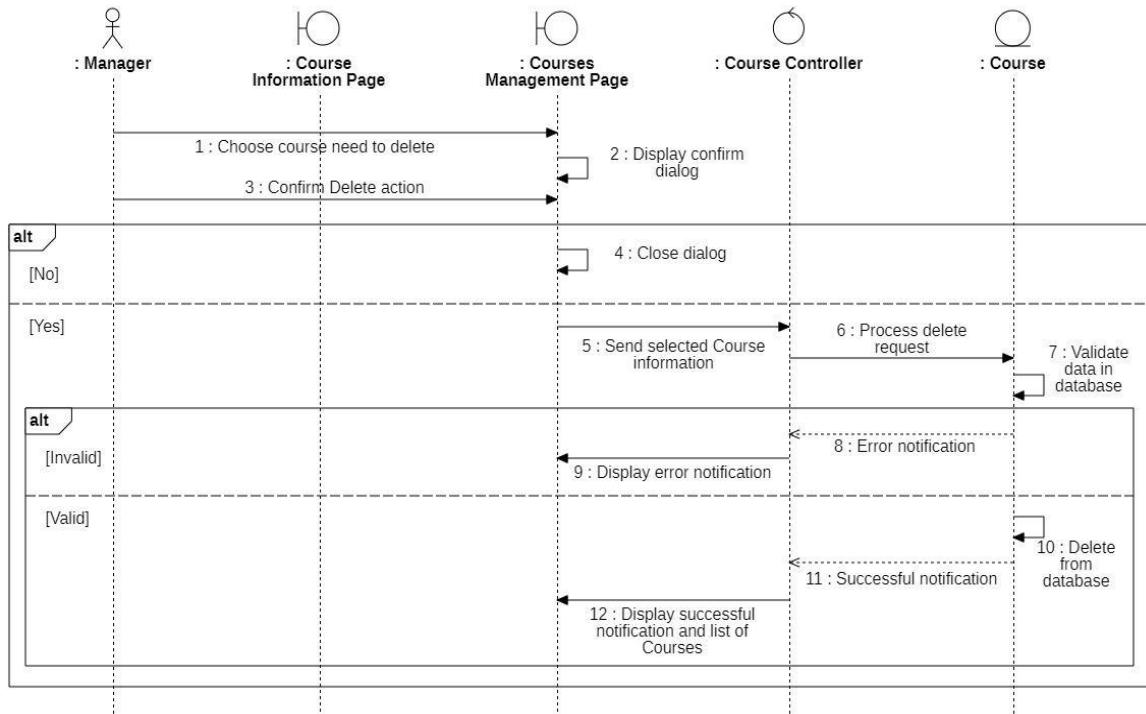


Figure 3. 31. Sequence Diagram for “Delete Course”

– Search

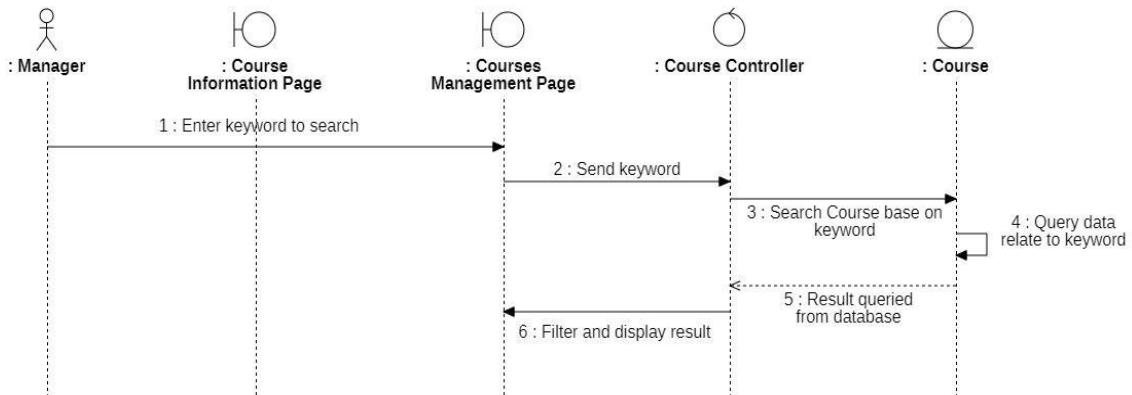


Figure 3. 32.Sequence Diagram for “Search Course”

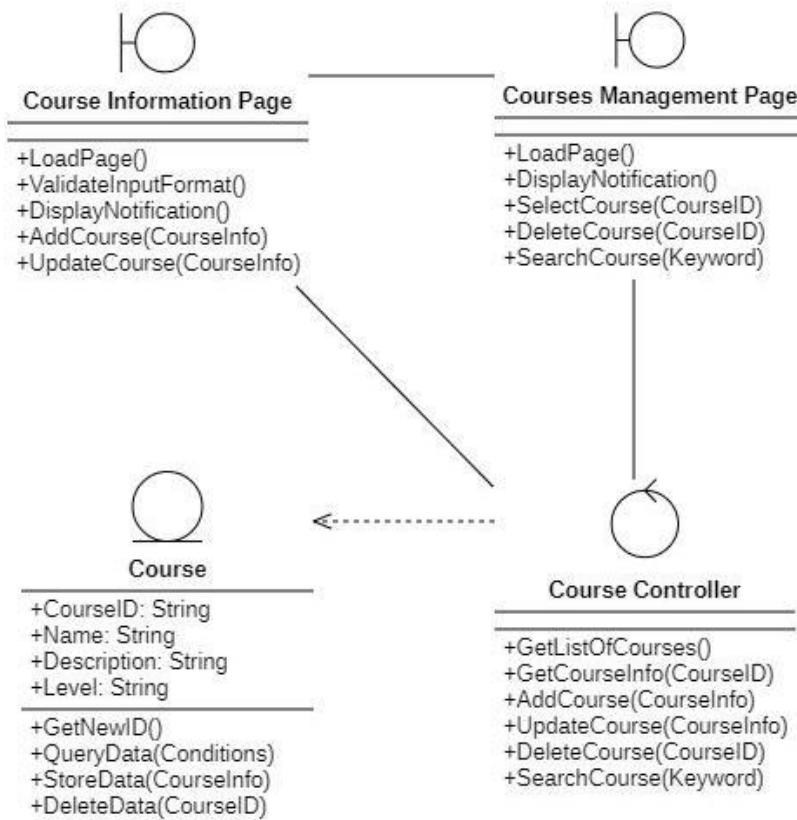


Figure 3. 33. Class Diagram for “Manage Course”

3.3.3. Manage Chapters

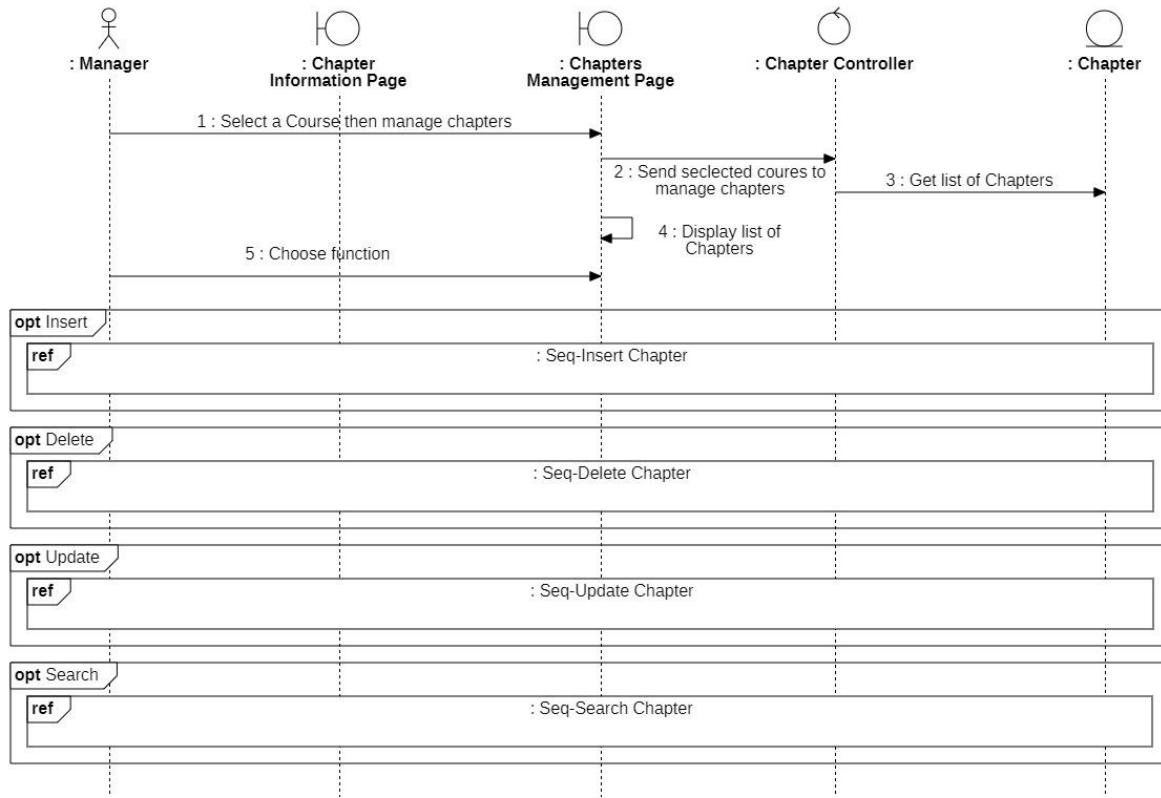


Figure 3. 34. Sequence Diagram for “Manage Chapters”

- Insert

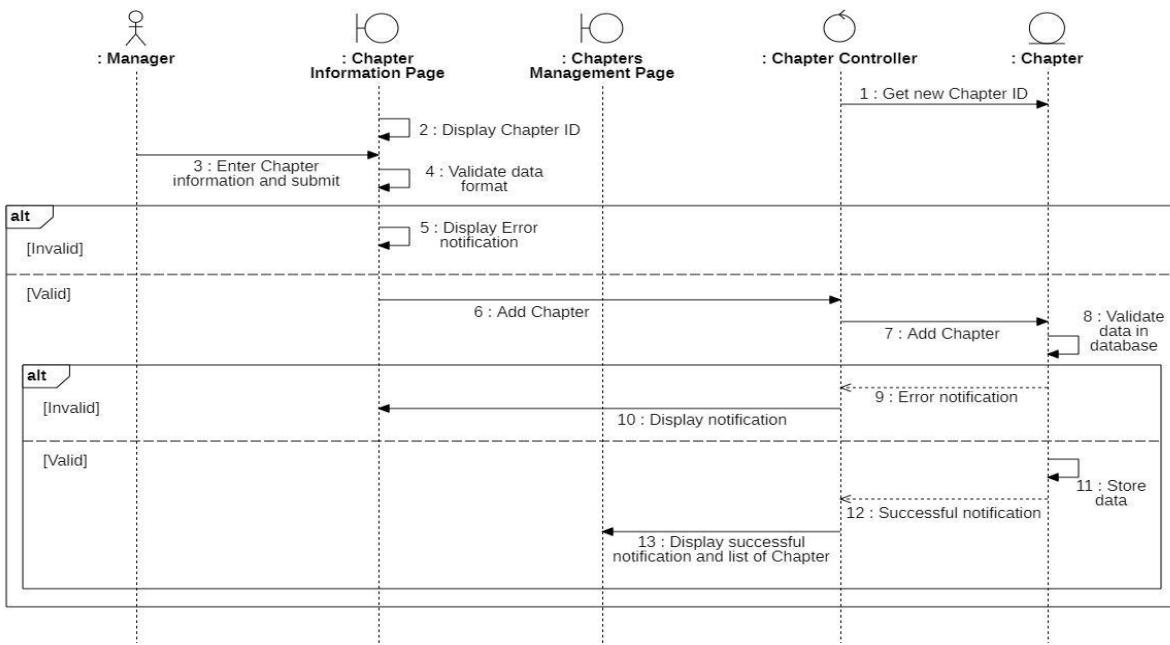


Figure 3. 35. Sequence Diagram for “Insert Chapter”

– Update

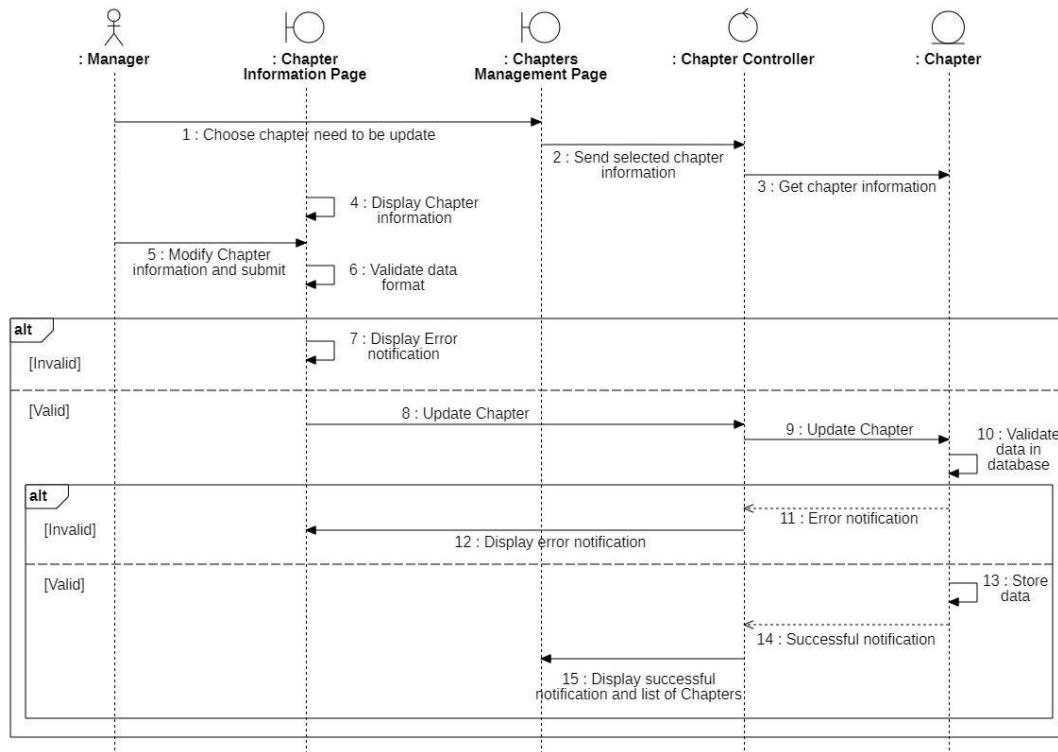


Figure 3. 36. Sequence Diagram for “Update Chapter”

– Delete

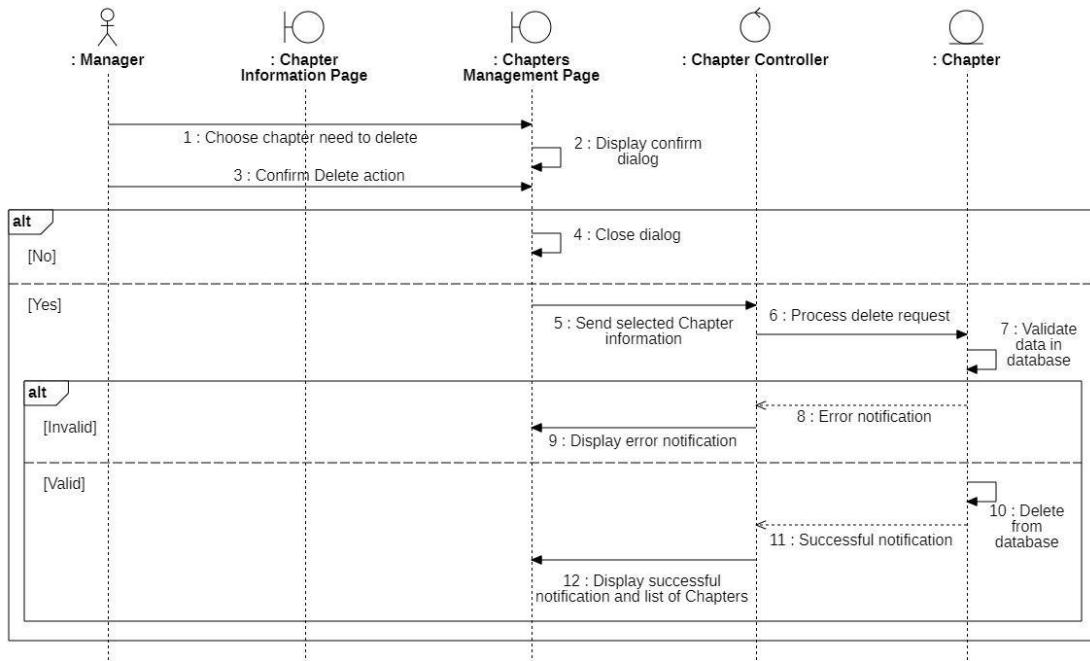


Figure 3. 37. Sequence Diagram for “Delete Chapter”

– Search

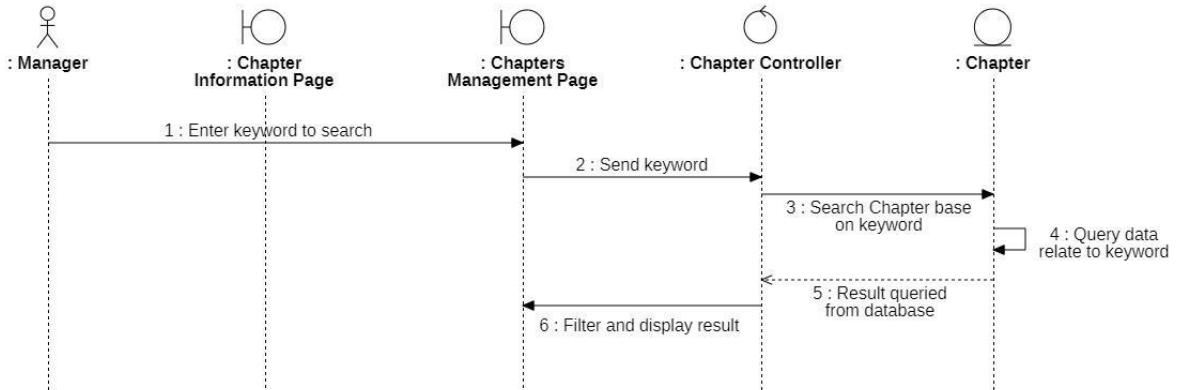


Figure 3. 38. Sequence Diagram for “Search Chapter”

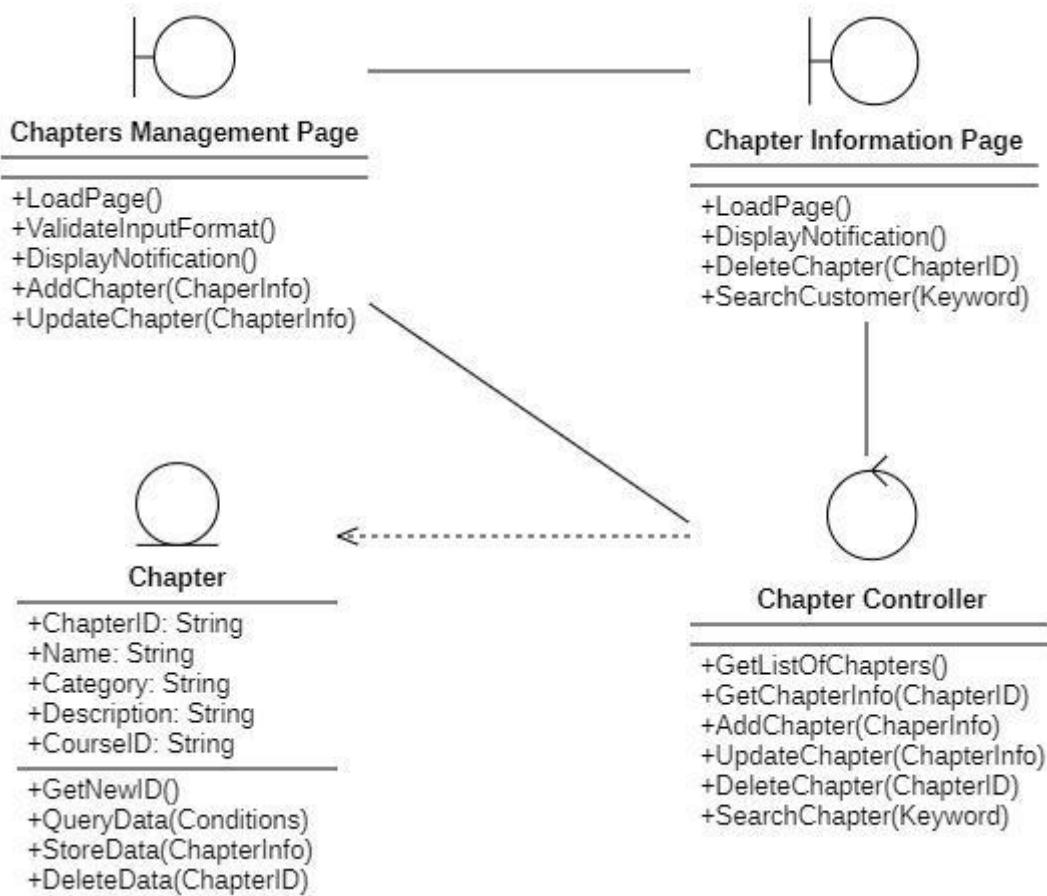


Figure 3. 39. Class Diagram for “Manage Chapter”

3.3.4. Manage Classes

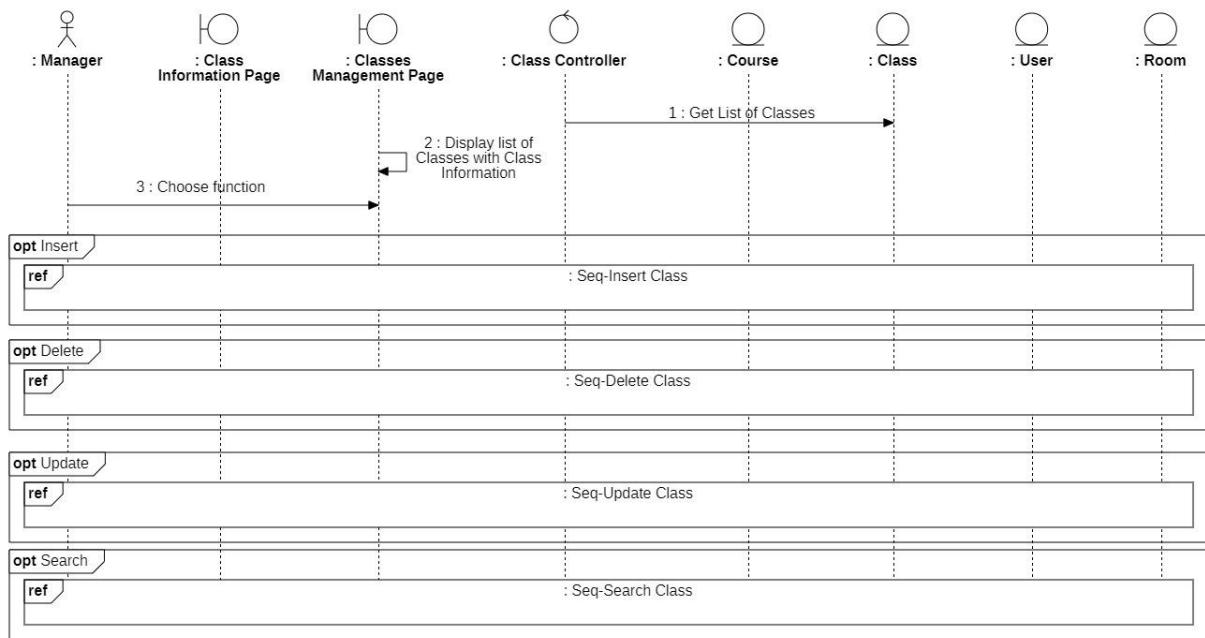


Figure 3. 40. Sequence Diagram for “Manage Classes”

– Insert

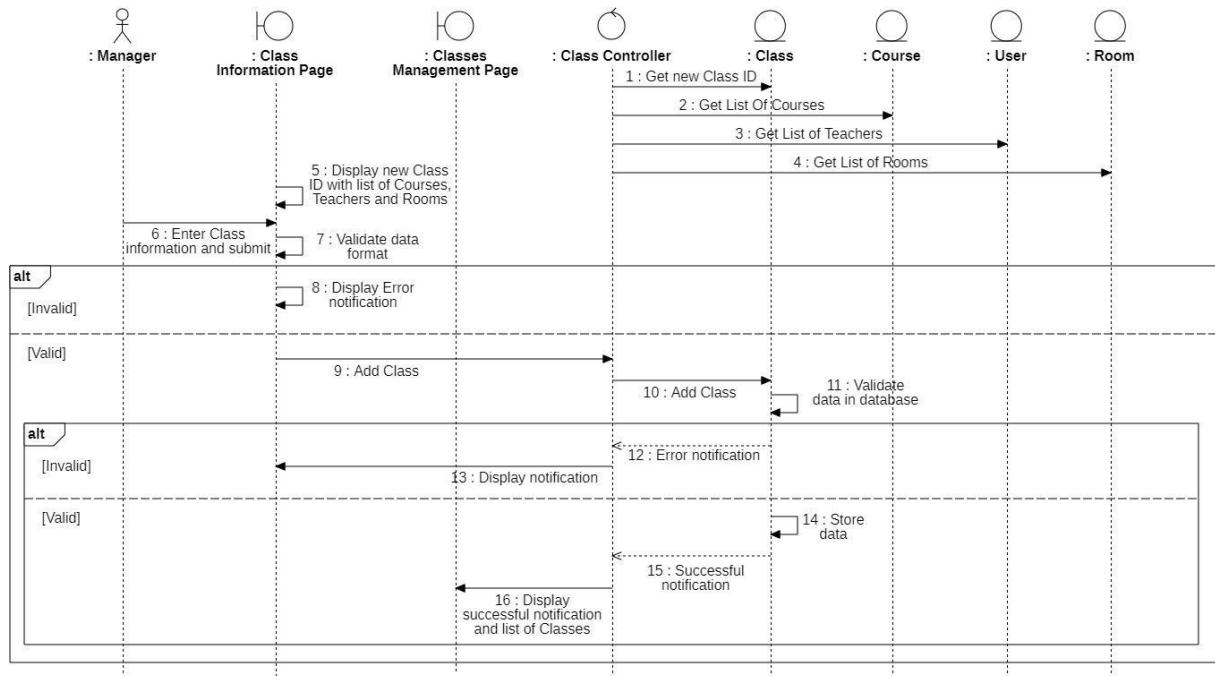


Figure 3. 41. Sequence Diagram for “Insert Class”

– Update

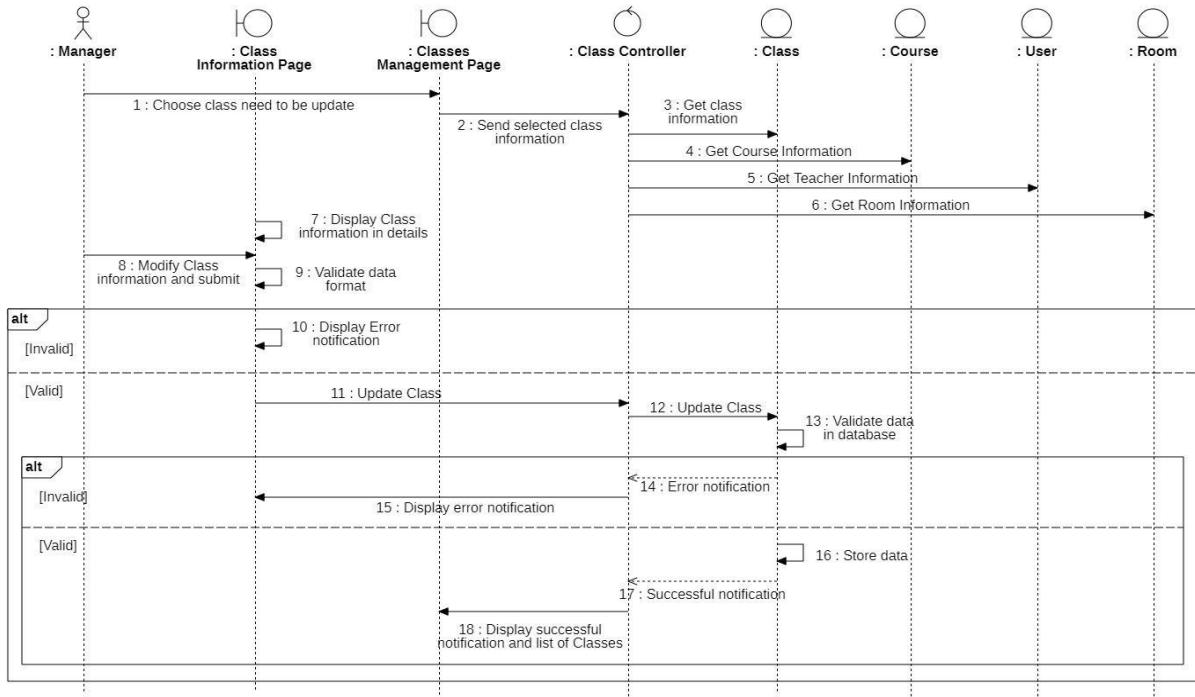


Figure 3. 42. Sequence Diagram for “Update Class”

– Delete

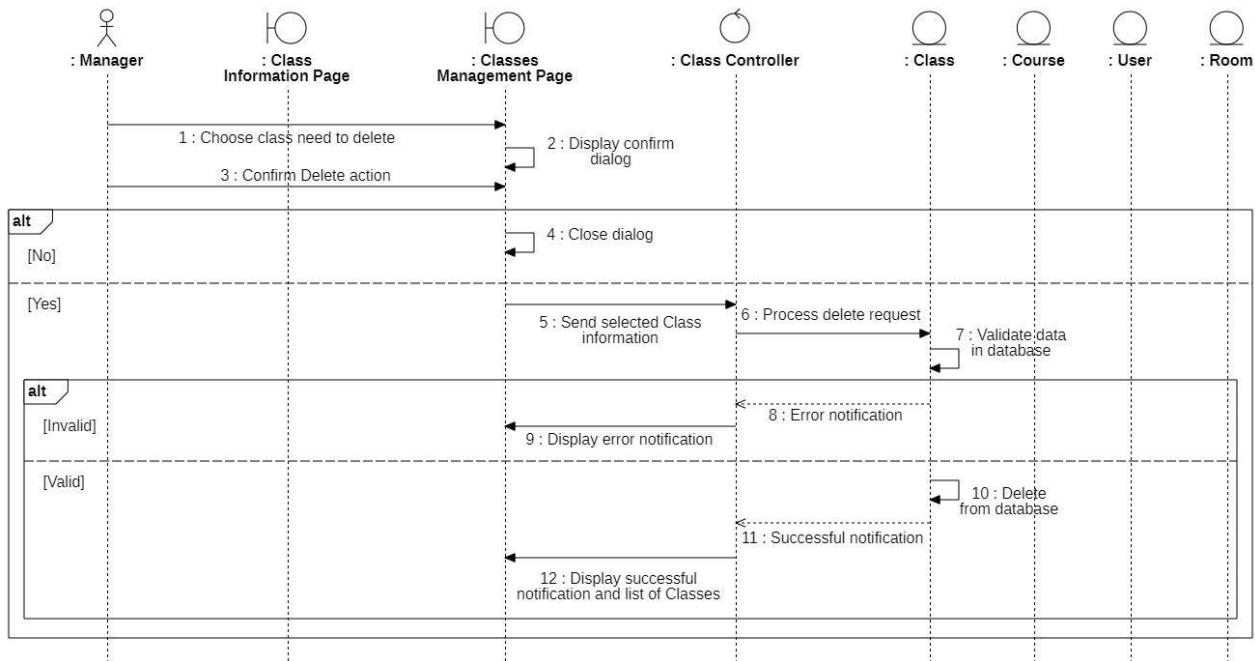


Figure 3. 43. Sequence Diagram for “Delete Class”

– Search

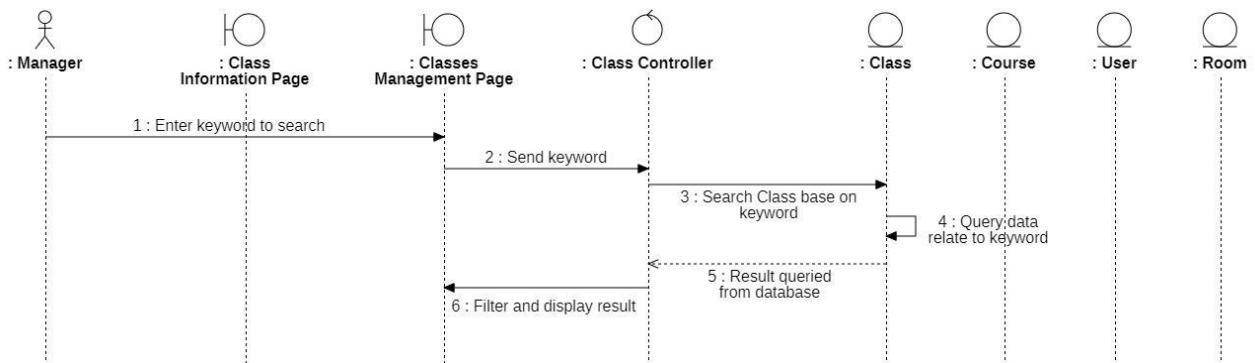


Figure 3. 44. Sequence Diagram for “Search Class”

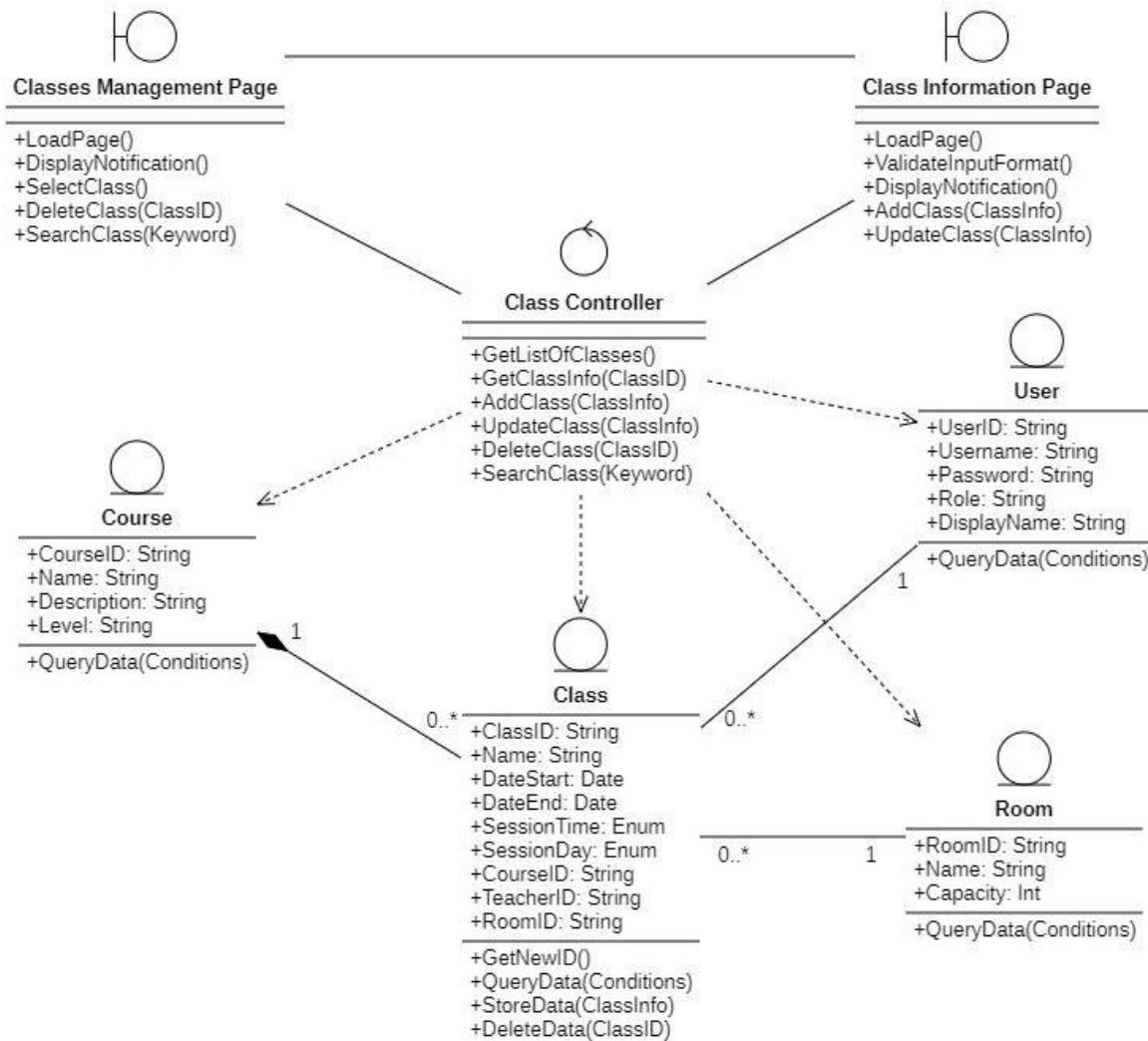


Figure 3. 45. Class Diagram for “Manage Class”

3.3.5. Manage Lessons

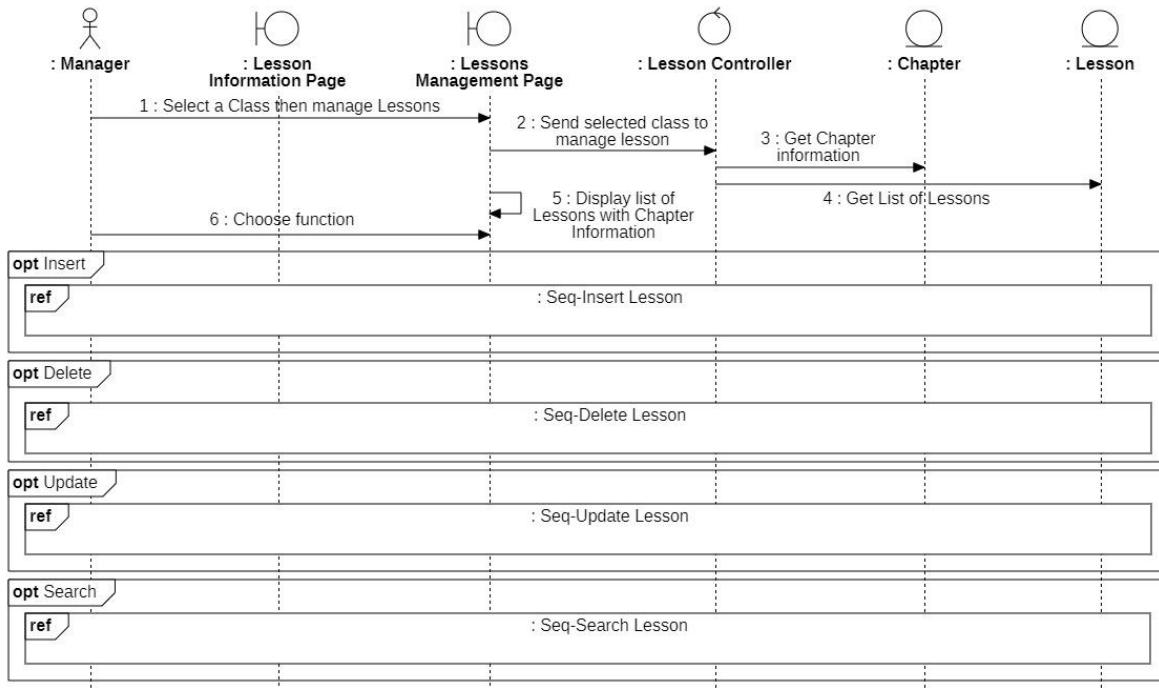


Figure 3. 46. Sequence Diagram for “**Manage Lessons**”

– Insert

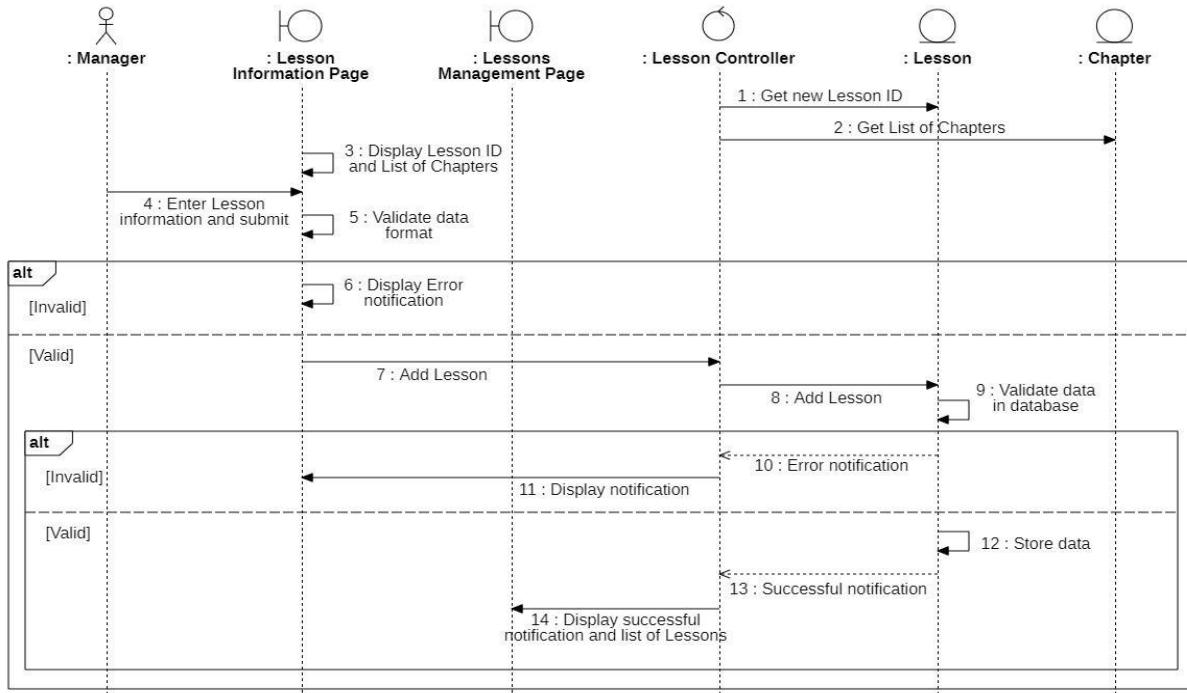


Figure 3. 47. Sequence Diagram for “Insert Lesson”

– Update

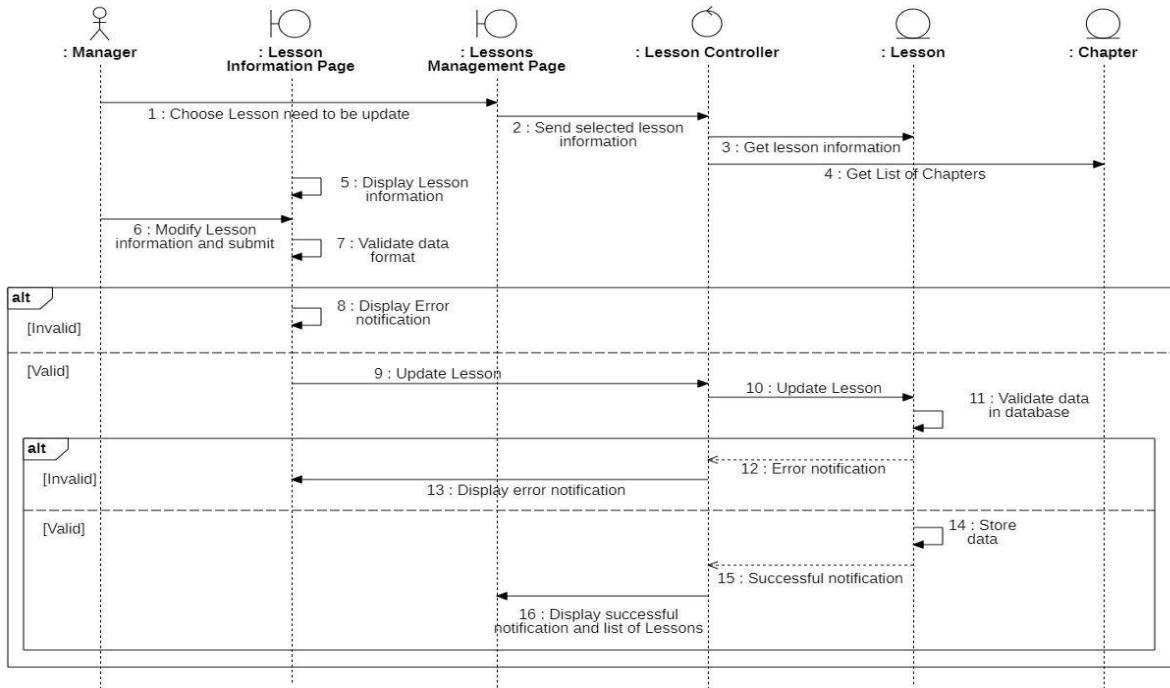


Figure 3. 48. Sequence Diagram for “Update Lesson”

– Delete

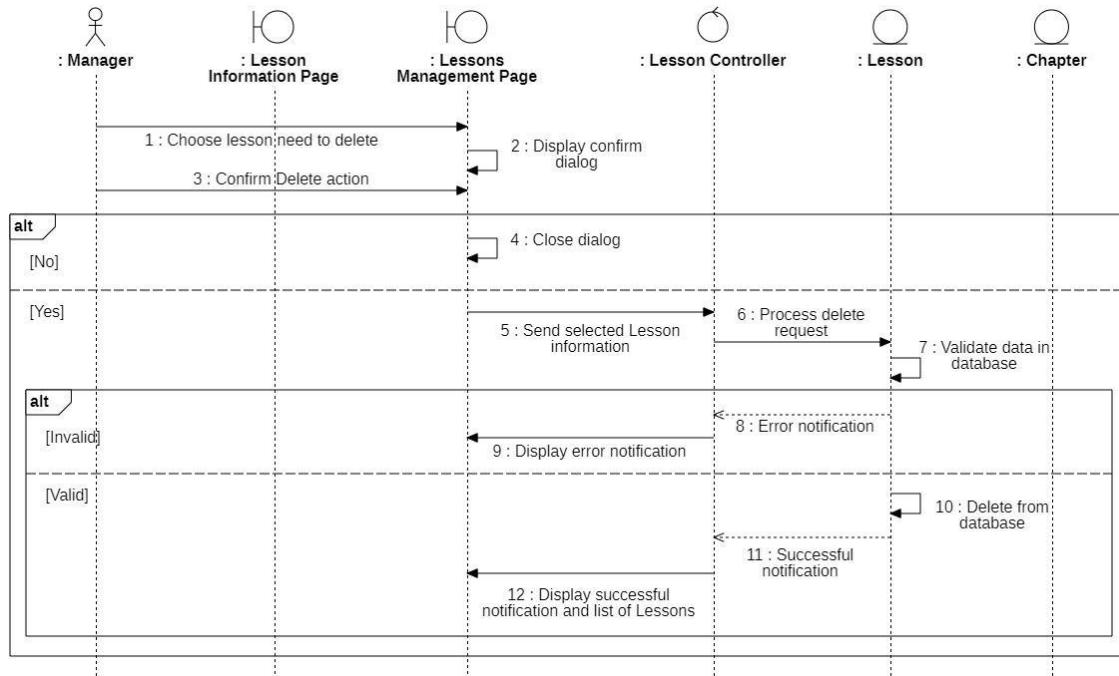


Figure 3. 49. Sequence Diagram for “Delete Lesson”

– Search

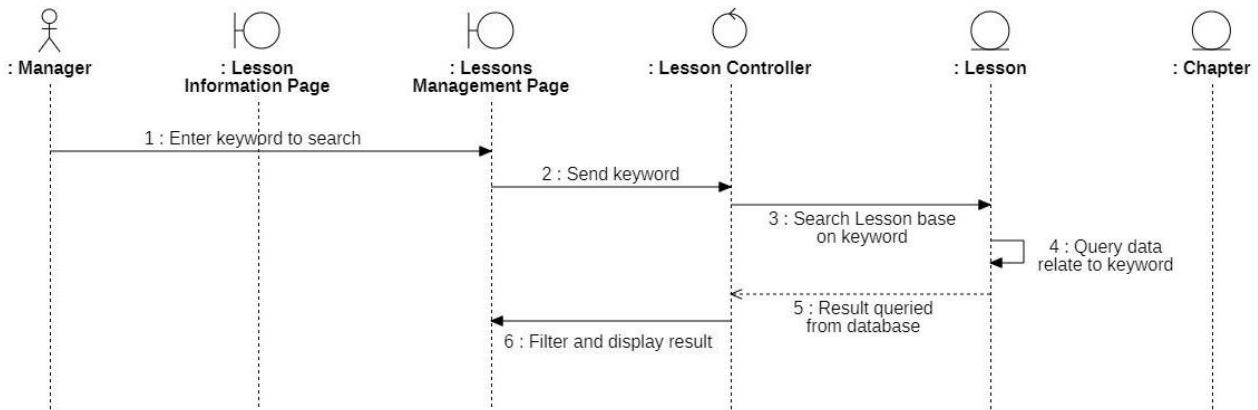


Figure 3. 50. Sequence Diagram for “Search Lesson”

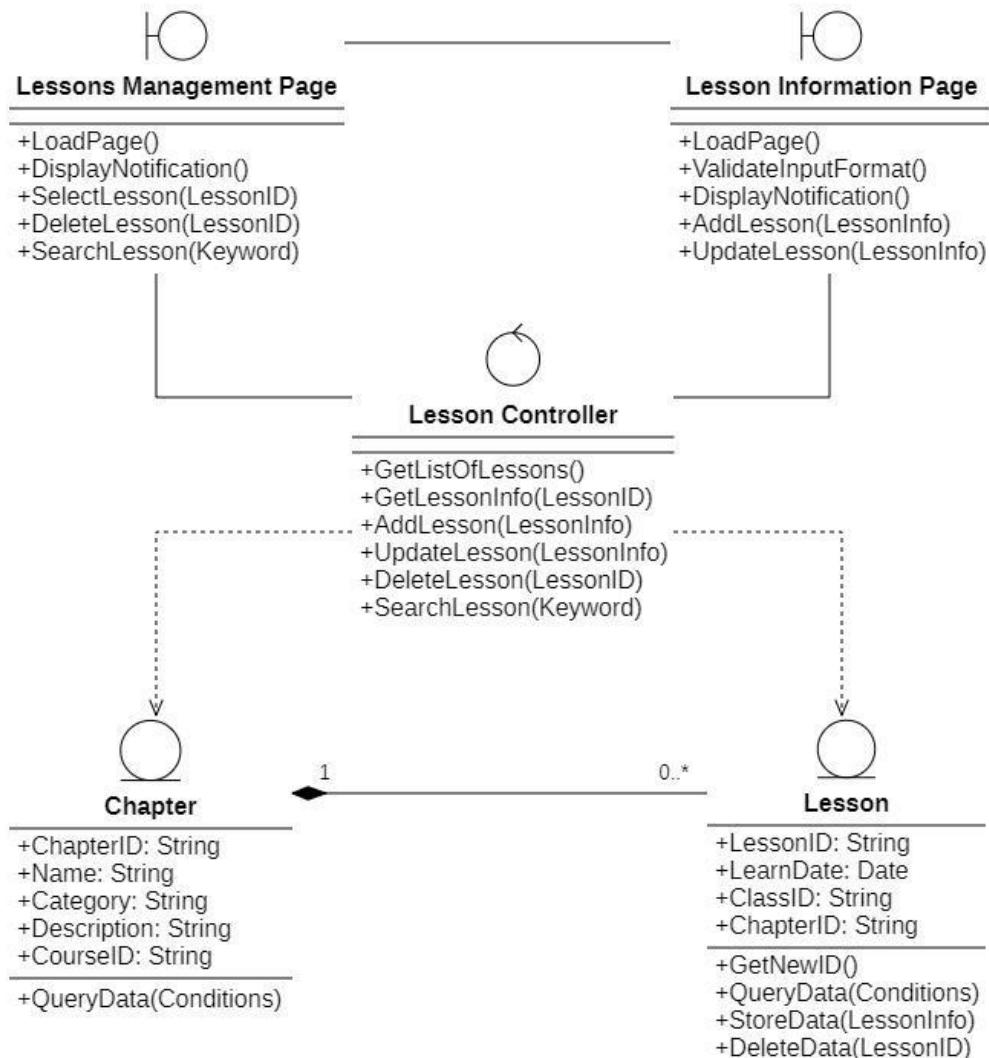


Figure 3. 51. Class Diagram for “Manage Lessons”

3.3.6. Manage Materials

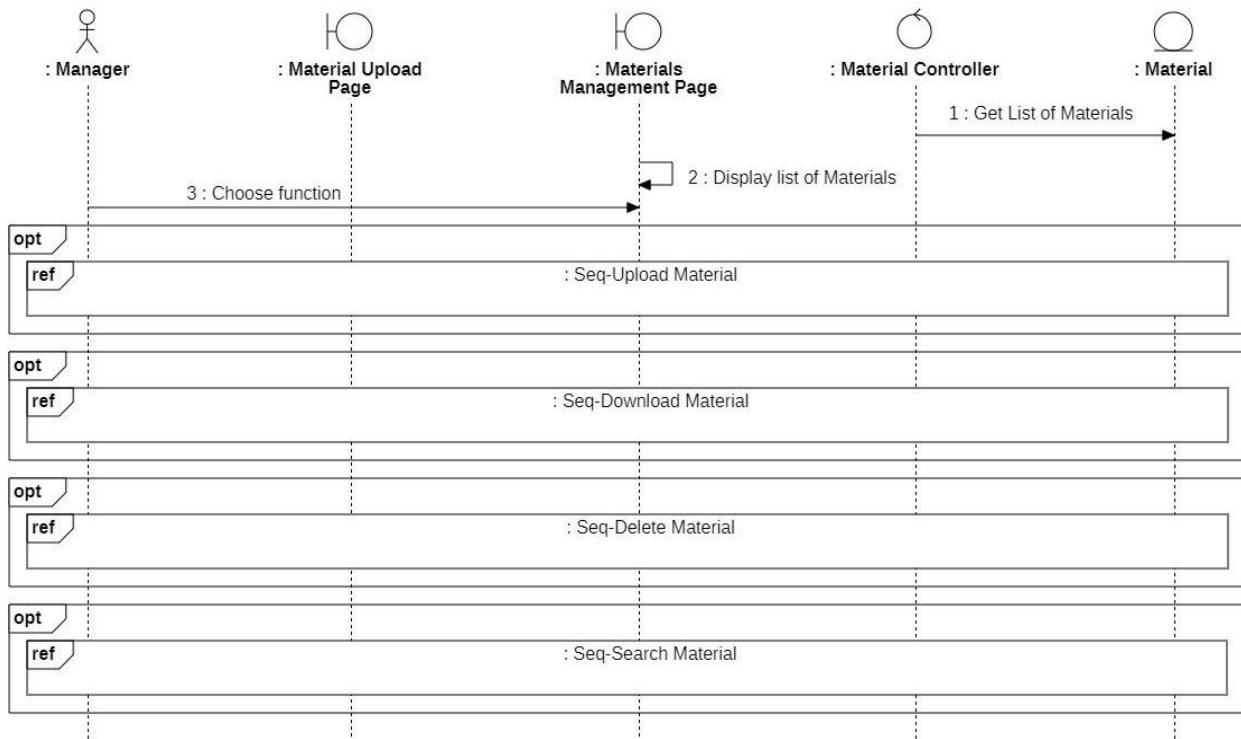


Figure 3. 52. Sequence Diagram for “Manage Materials”

– Upload

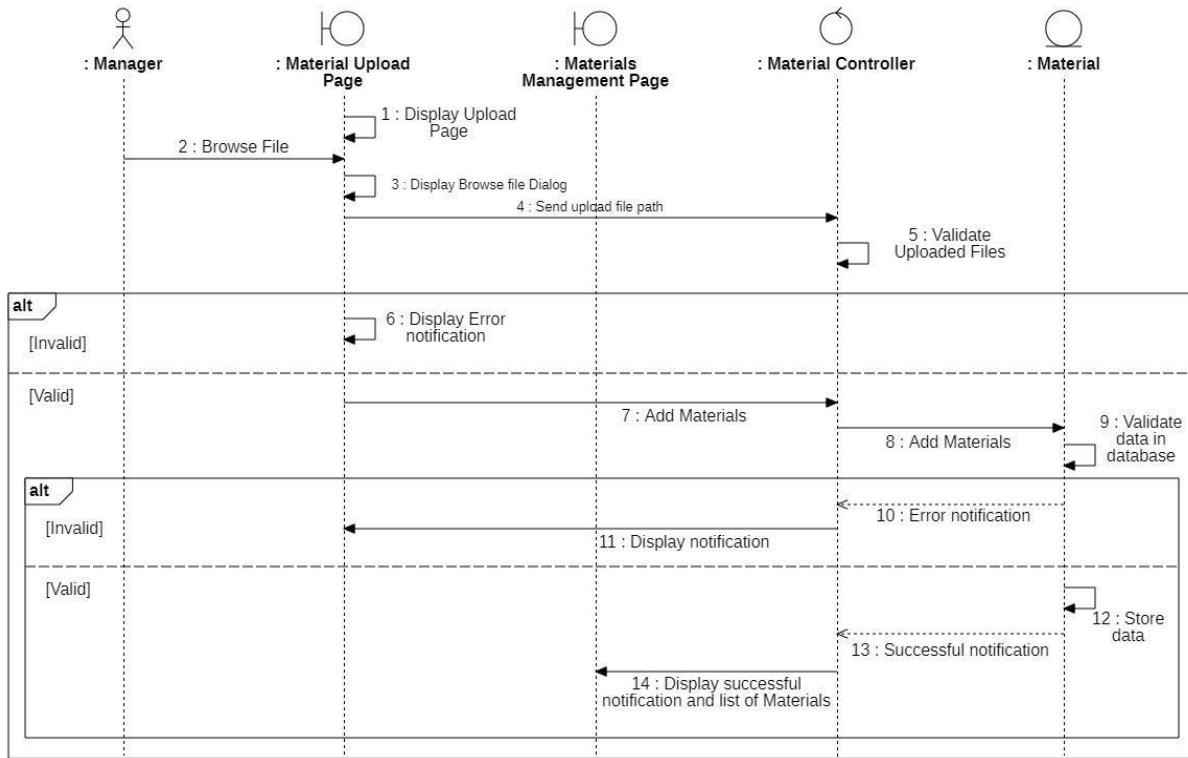


Figure 3. 53. Sequence Diagram for “Upload Materials”

– Download

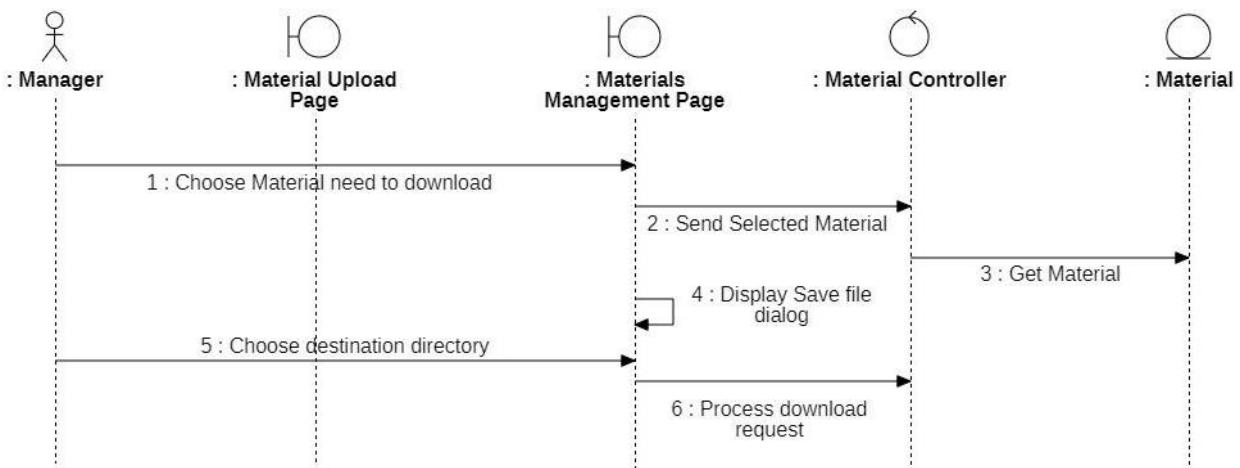


Figure 3. 54. Sequence Diagram for “Download Material”

– Delete

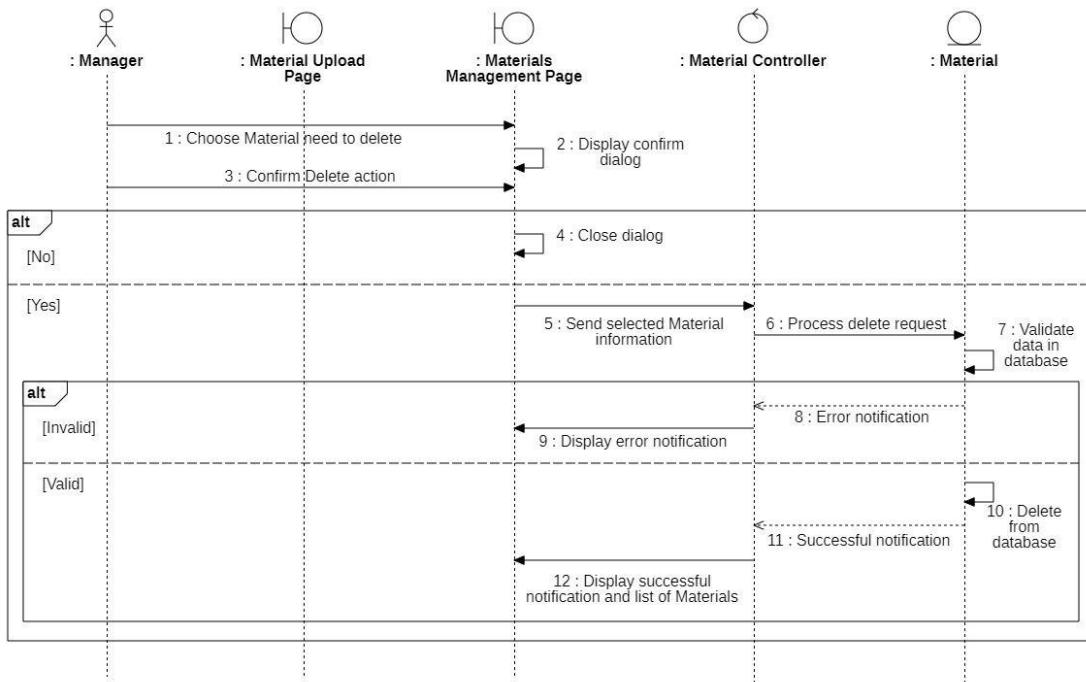


Figure 3. 55. Sequence Diagram for “Delete Material”

– Search

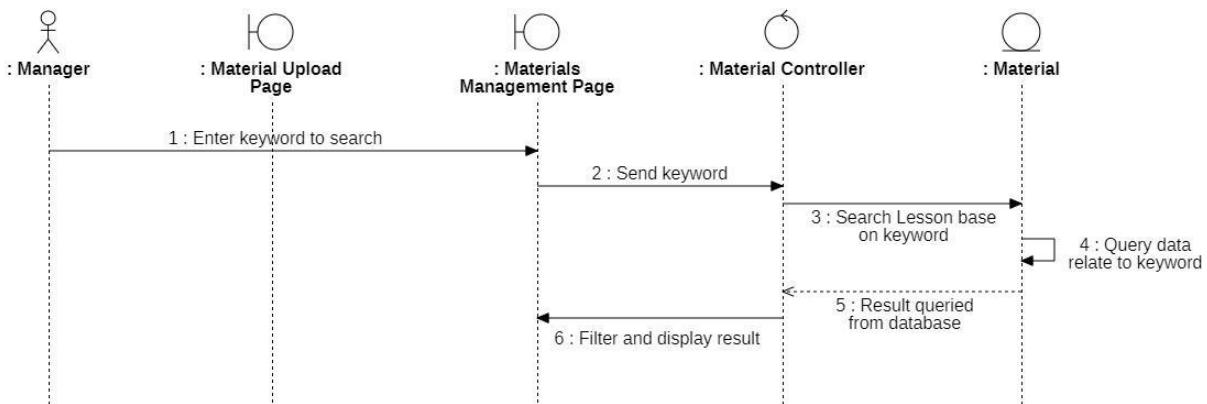
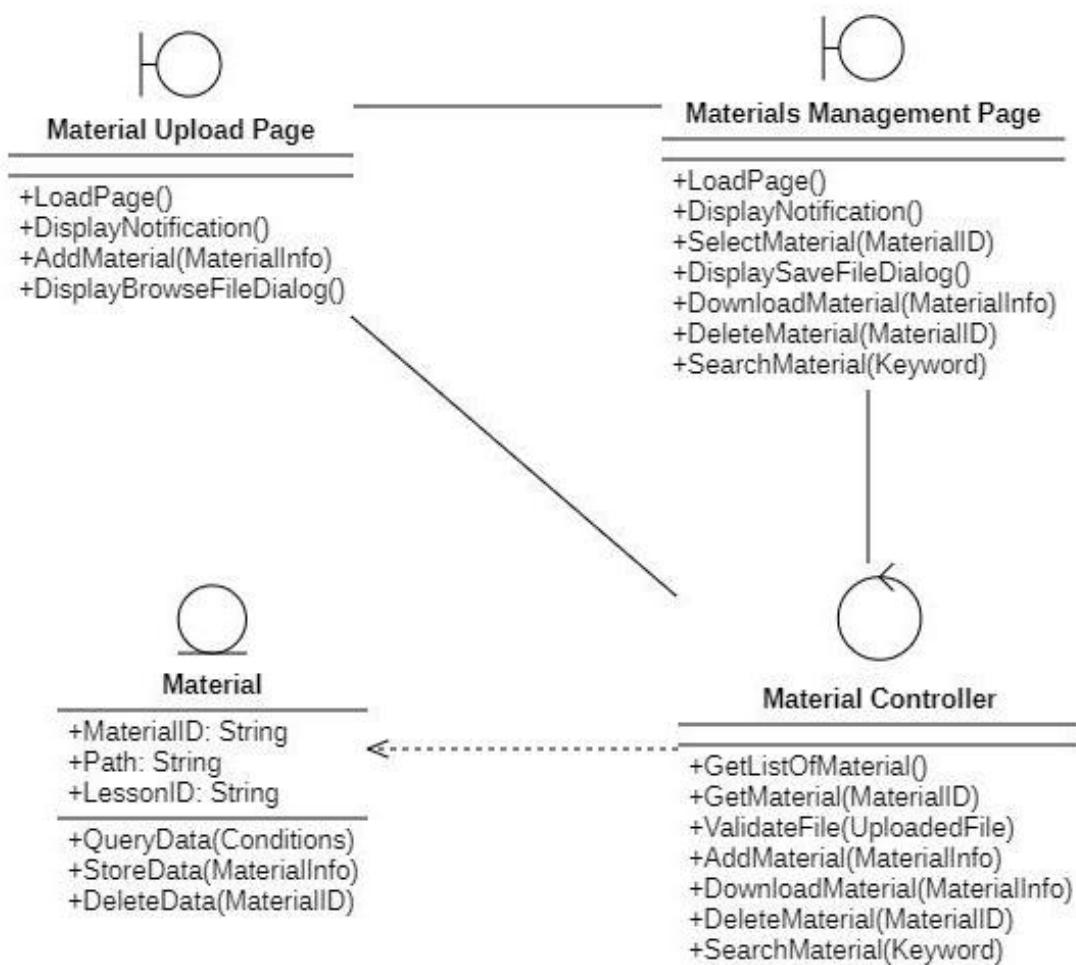


Figure 3. 56. Sequence Diagram for “Search Material”

Figure 3. 57. Class Diagram for “**Manage Materials**”

3.3.7. Manage Attendances

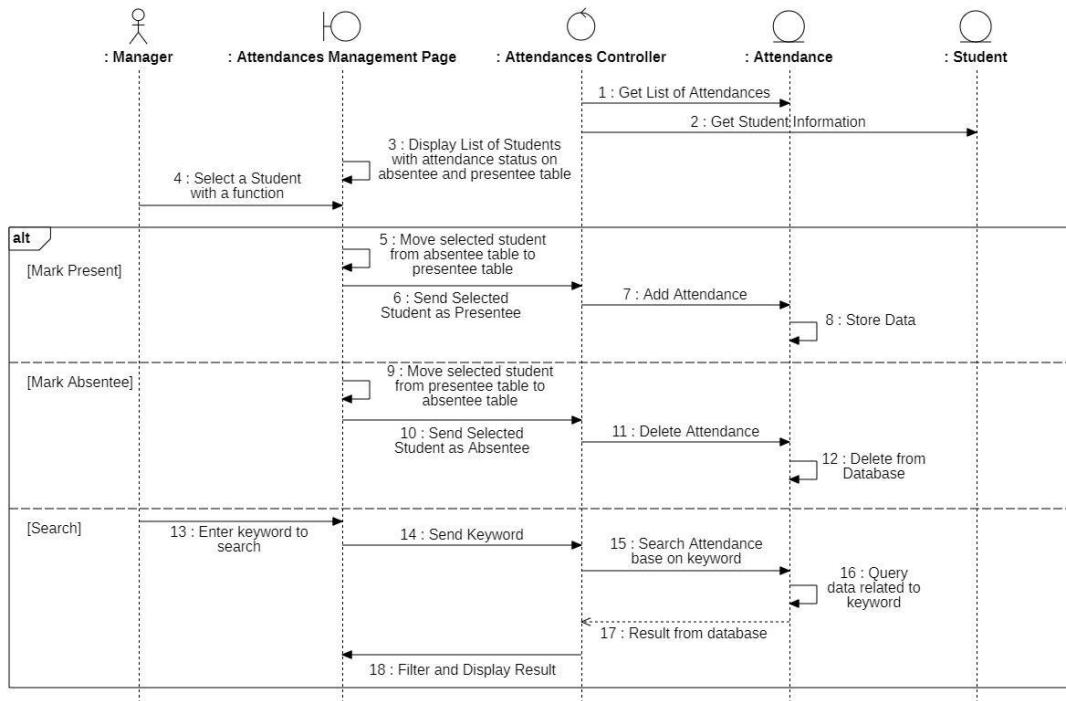


Figure 3. 58. Class Diagram for “Manage Attendances”

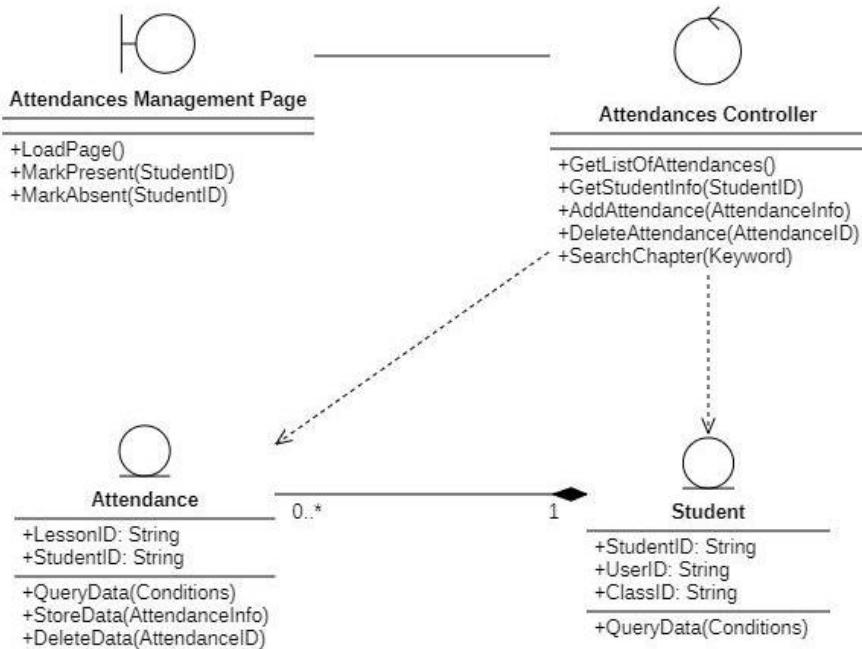


Figure 3. 59. Class Diagram for “Manage Attendances”

3.3.8. Manage Classrooms

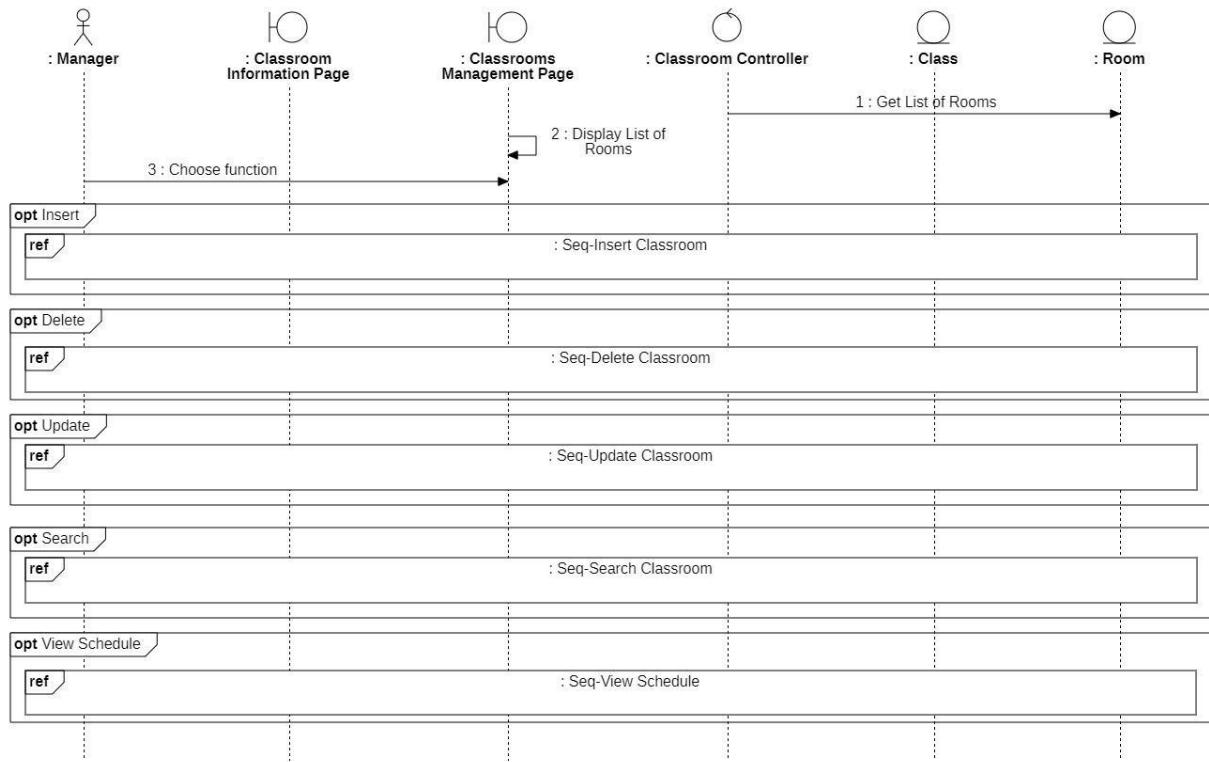


Figure 3. 60. Sequence Diagram for “Manage Classrooms”

– Insert

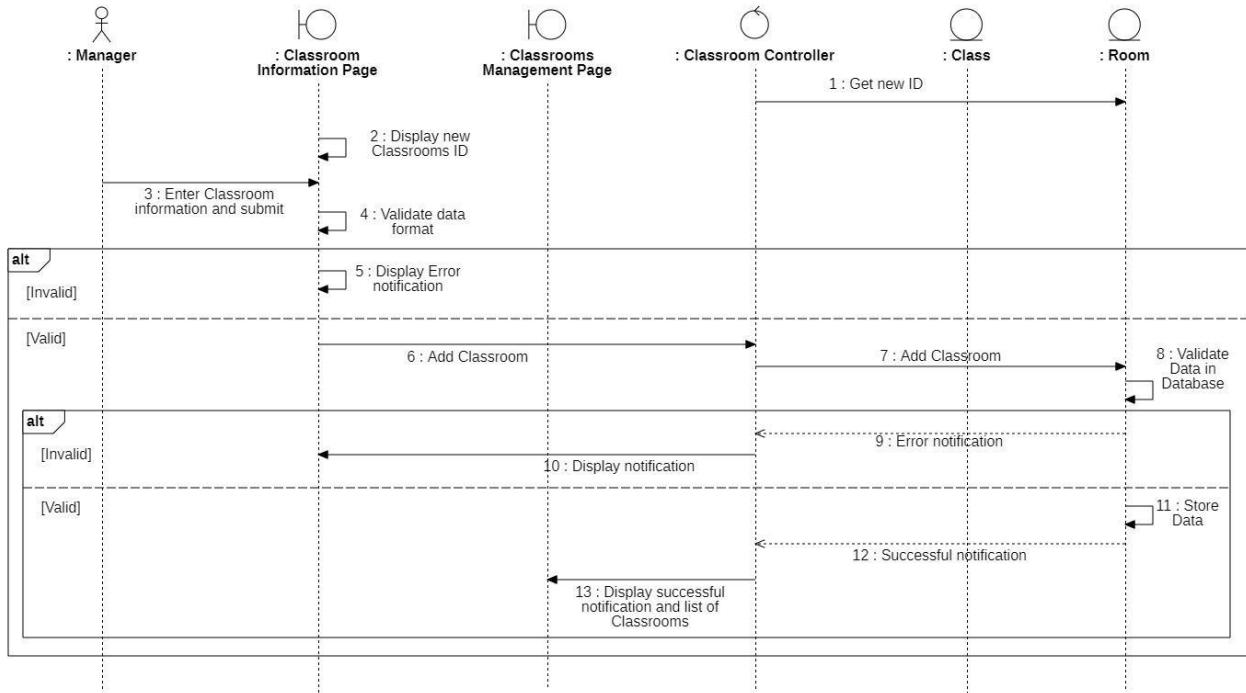


Figure 3. 61. Sequence Diagram for “Insert Classrooms”

– Update

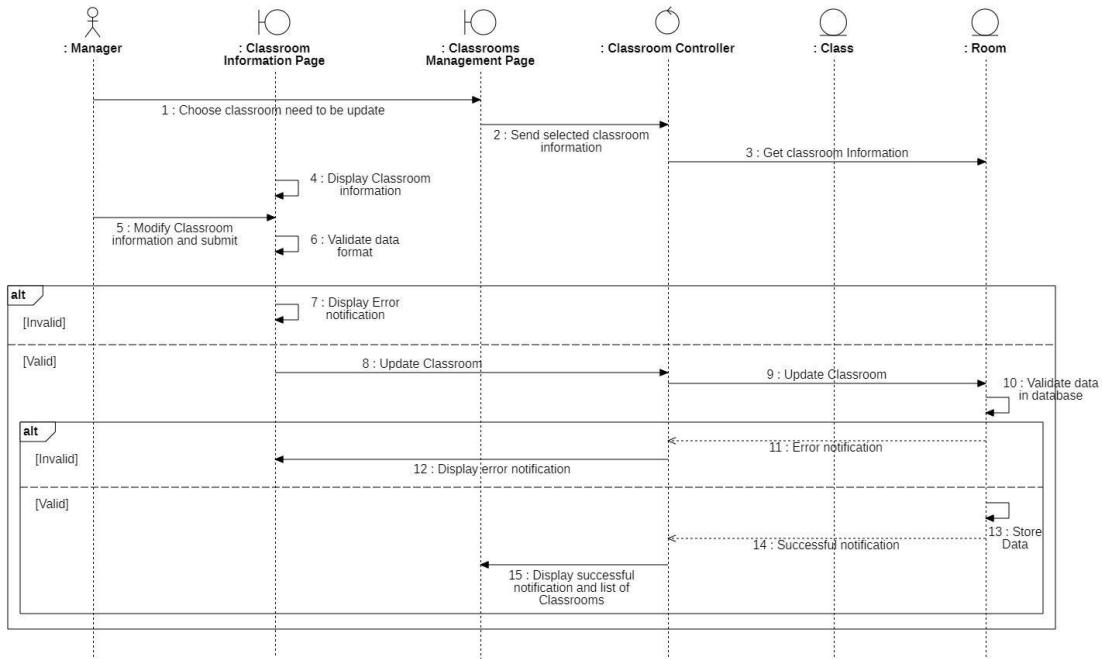


Figure 3. 62. Sequence Diagram for “Update Classrooms”

– Delete

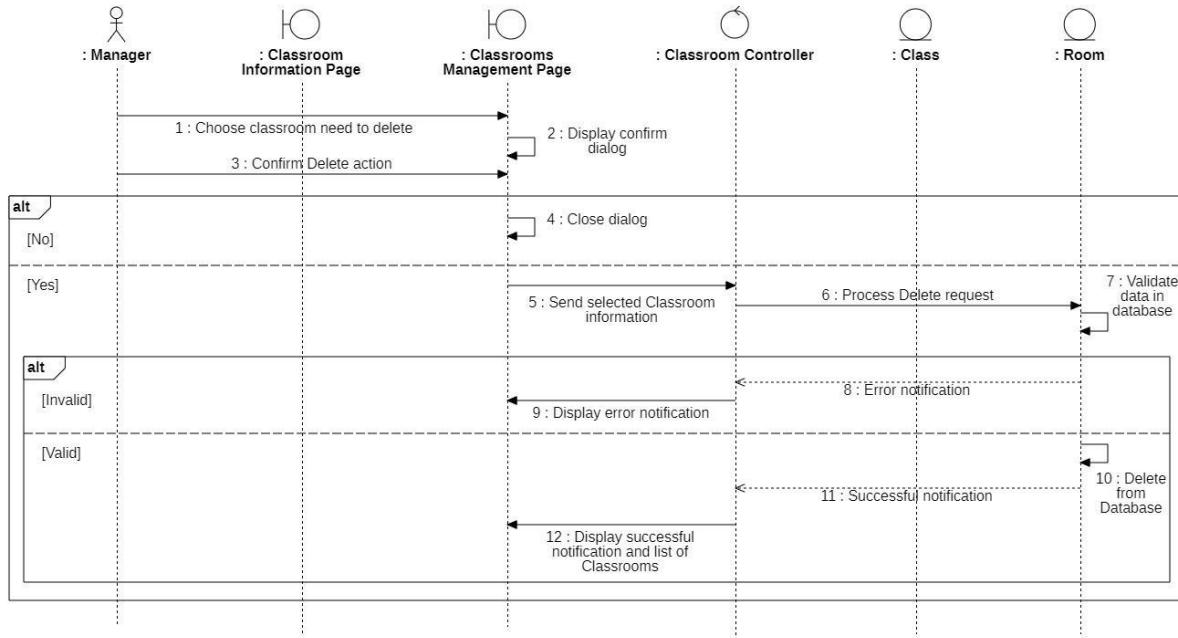


Figure 3. 63.Sequence Diagram for “Delete Classrooms”

– Search

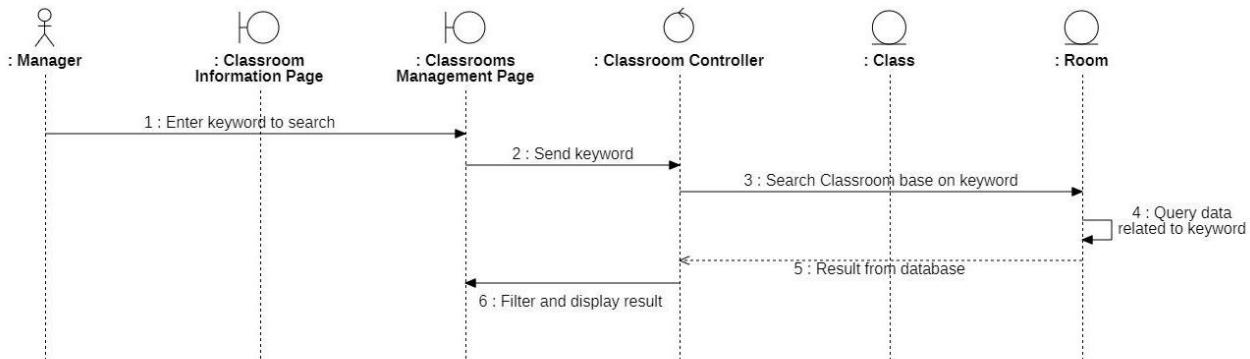


Figure 3. 64. Sequence Diagram for “Search Classrooms”

– View Schedule

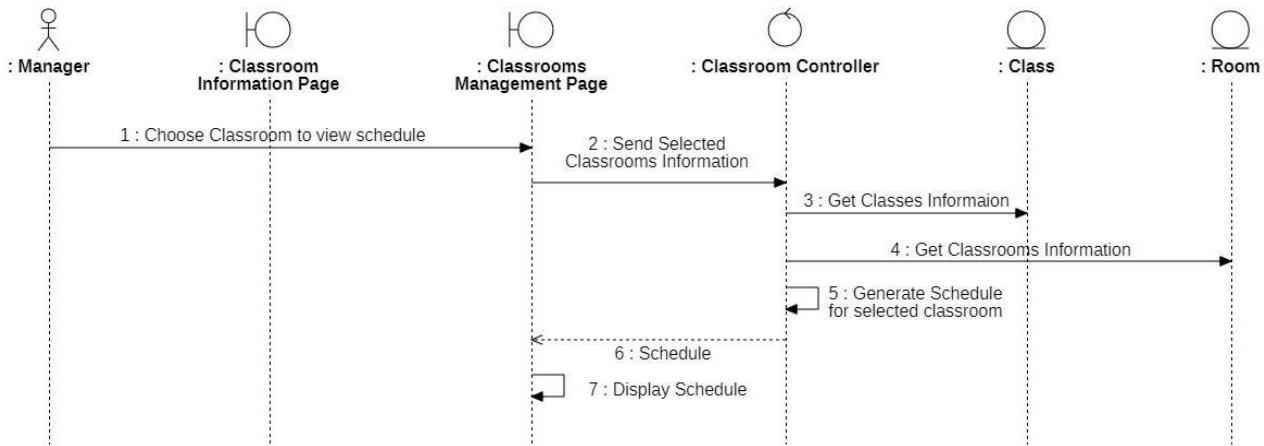


Figure 3. 65.Sequence Diagram for “View Schedule”

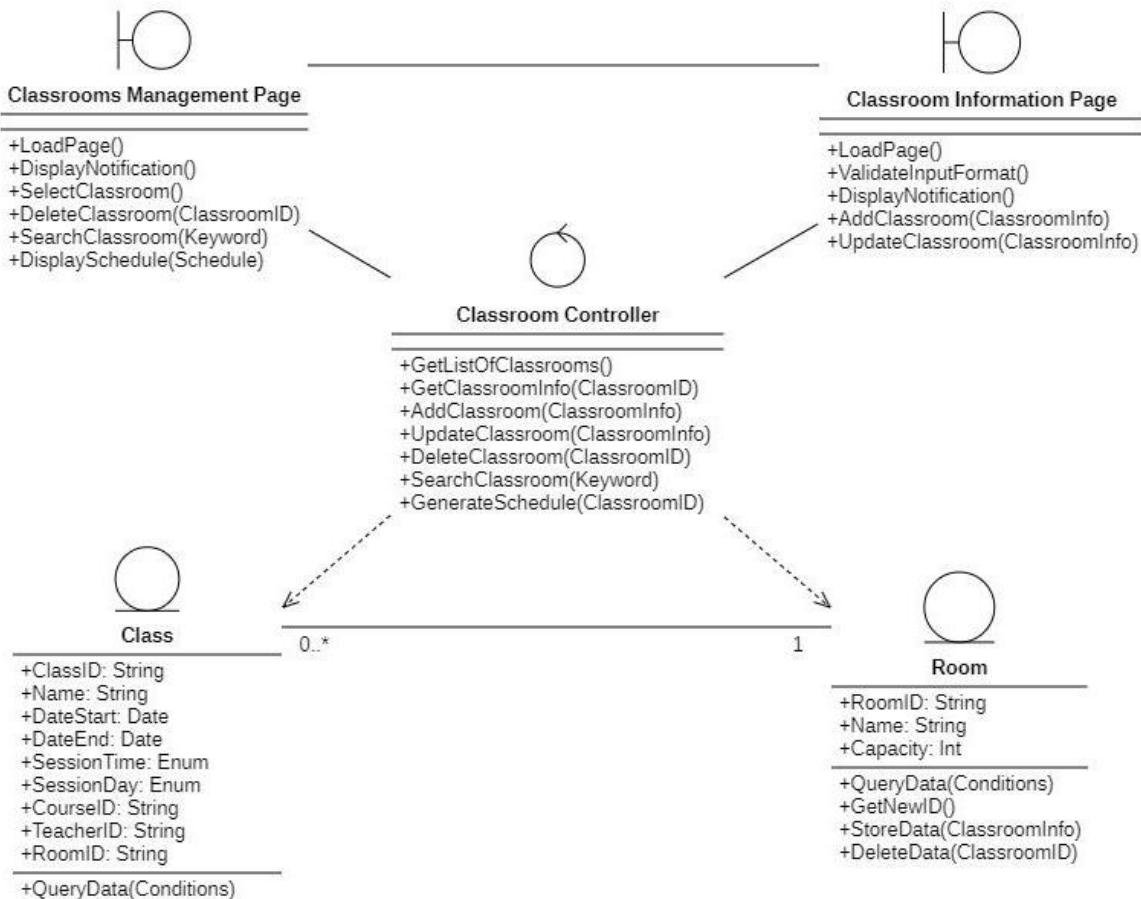


Figure 3. 66. Class Diagram for “Manage Classrooms”

3.3.9. Manage Users

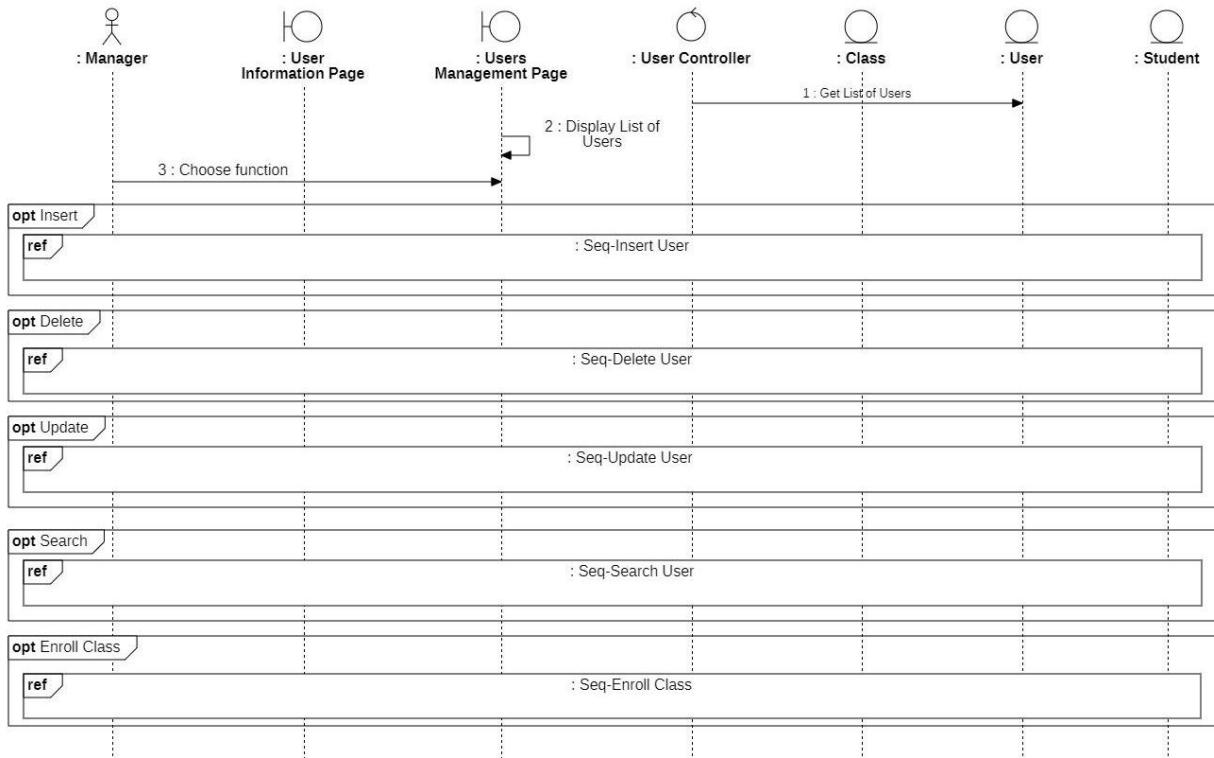
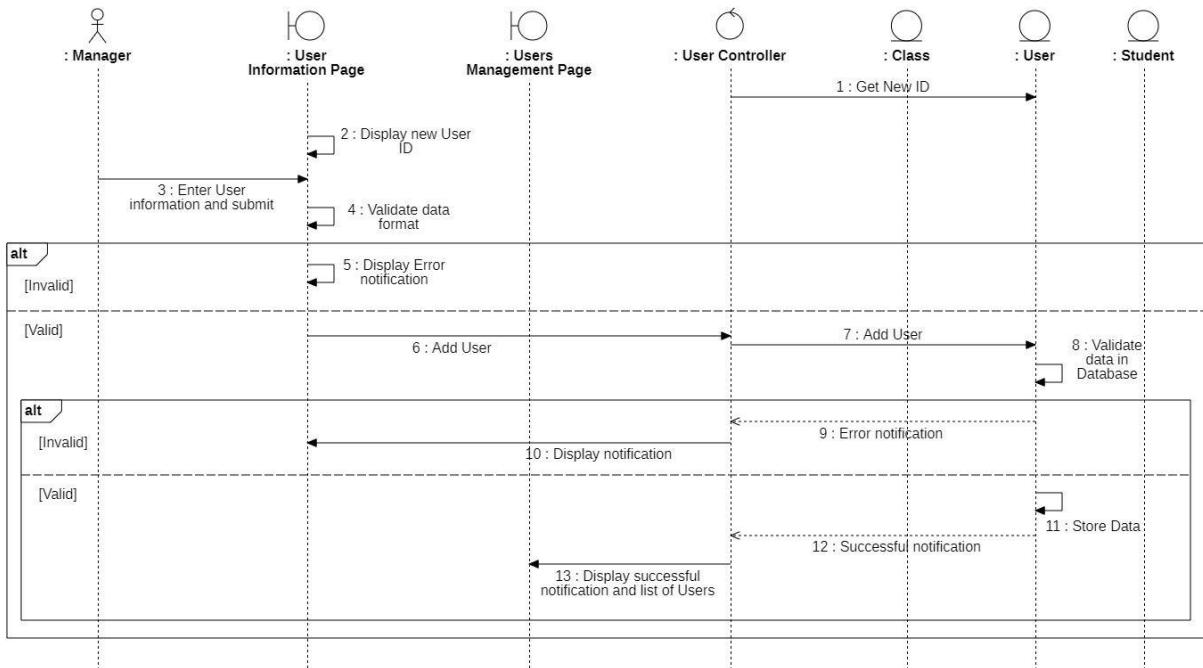
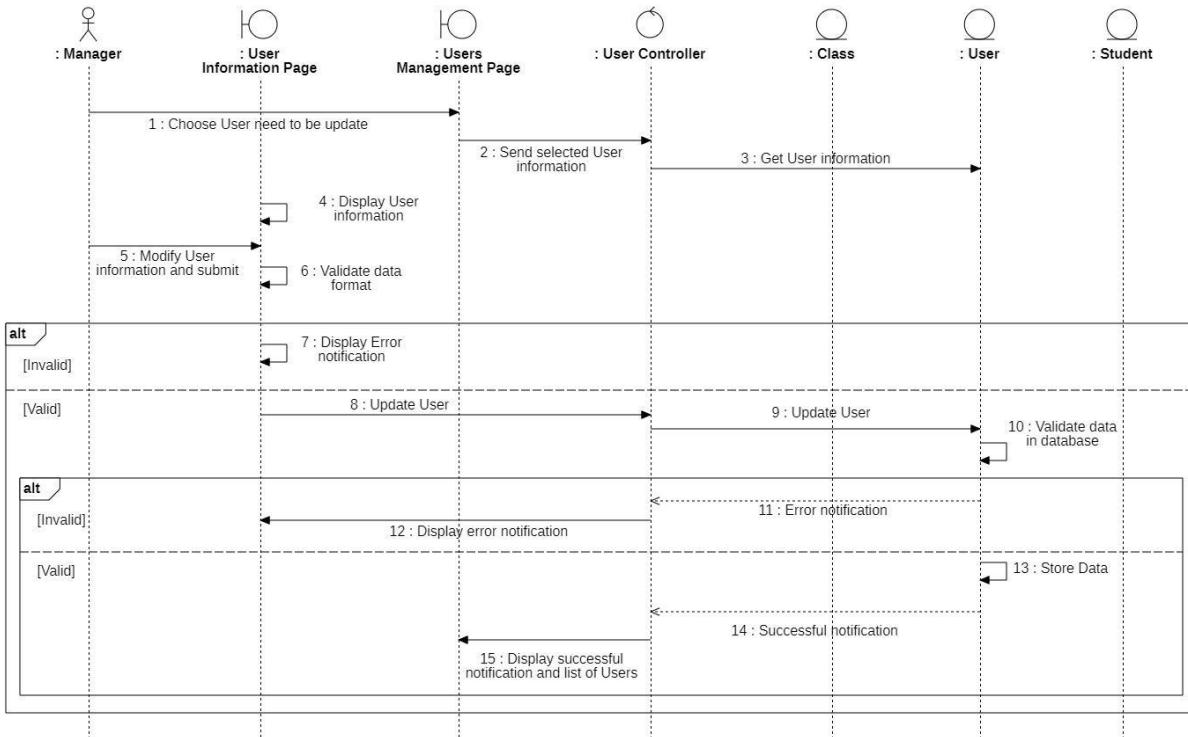


Figure 3. 67.Sequence Diagram for “Manage Users”

– Insert

Figure 3. 68.Sequence Diagram for “**Insert User**”

– Update

Figure 3. 69.Sequence Diagram for “**Update Users**”

– Delete

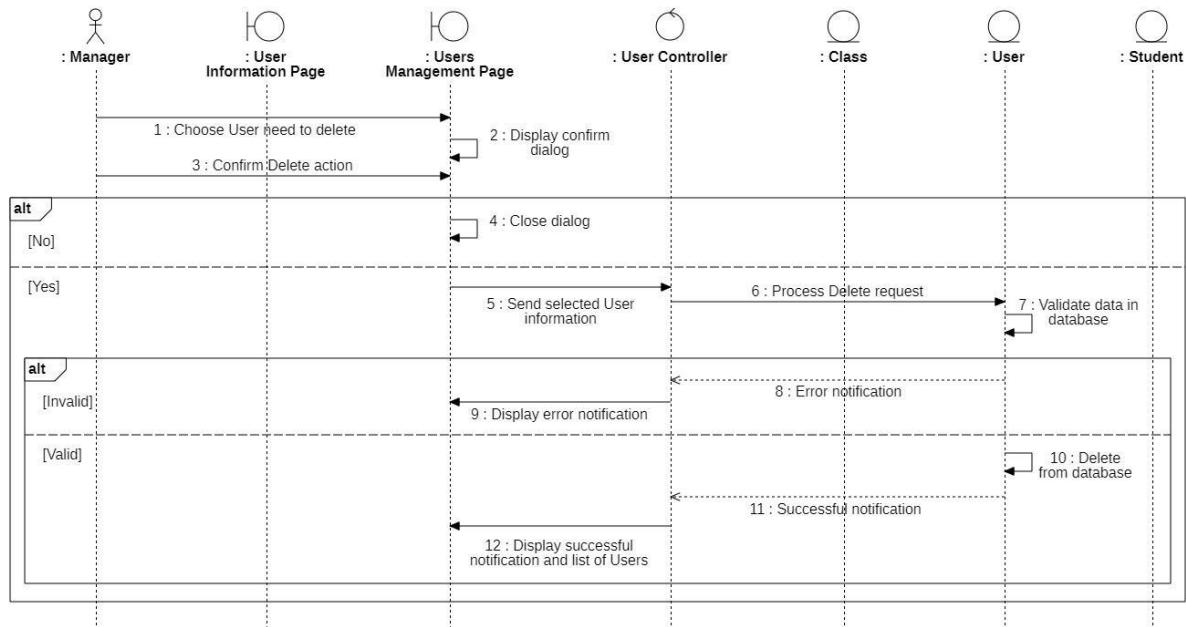


Figure 3. 70.Sequence Diagram for “Delete User”

– Search

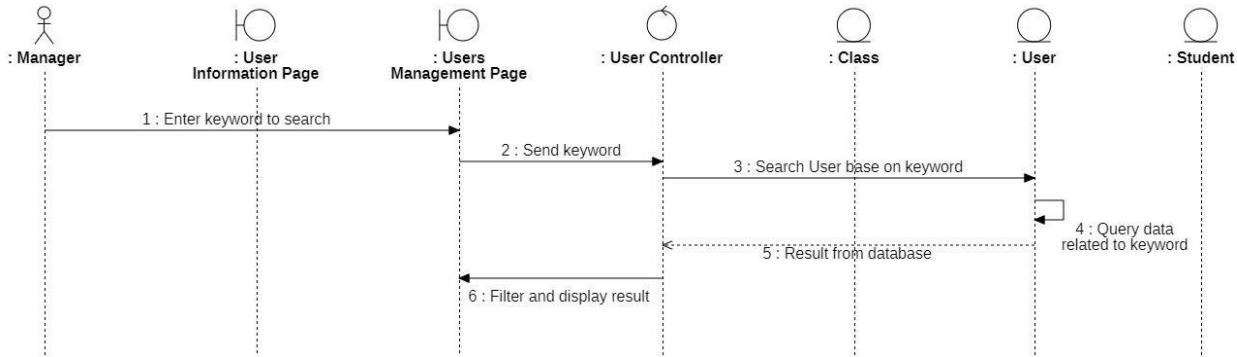


Figure 3. 71.Sequence Diagram for “Search Users”

- Enroll Class

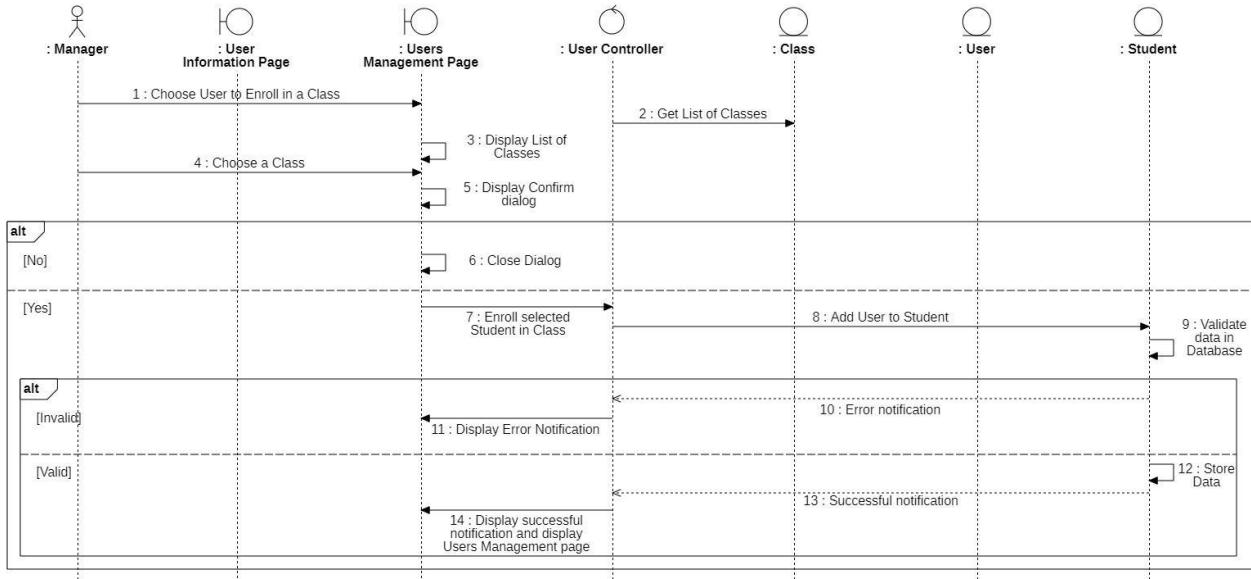


Figure 3. 72. Sequence Diagram for “Enroll Class”

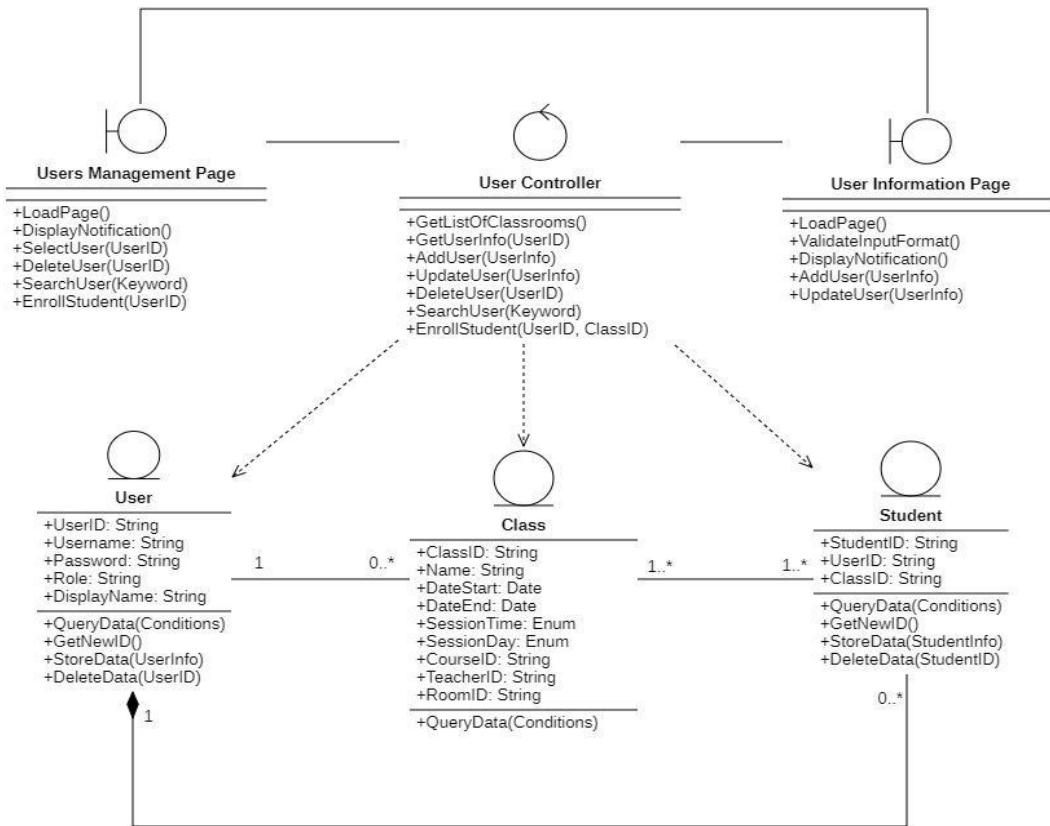


Figure 3. 73. Class Diagram for “Manage Users”

3.3.10. Export Class Report

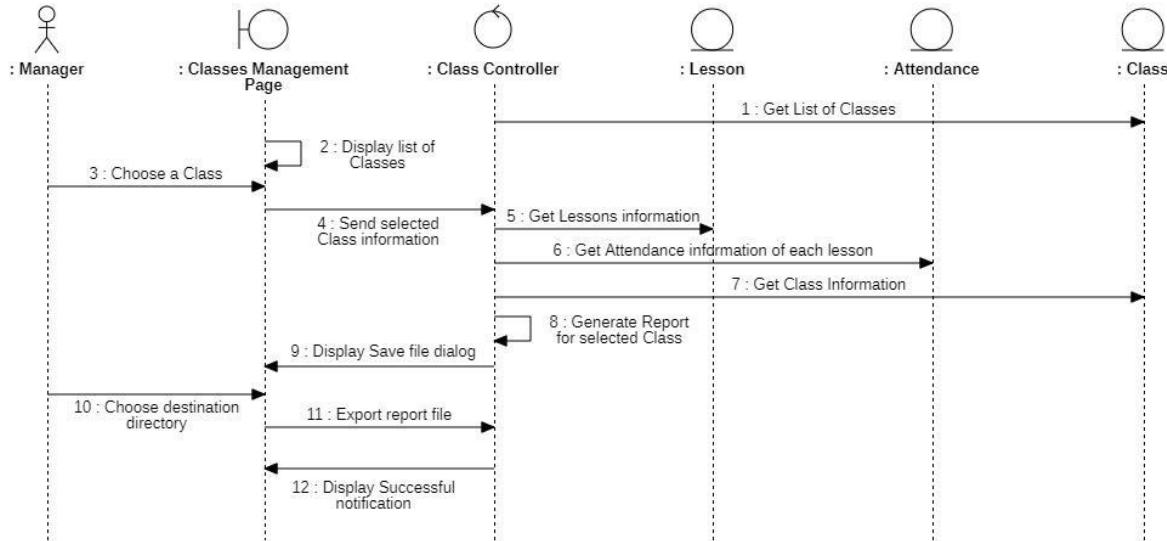


Figure 3. 74. Sequence Diagram for “Export Class Report”

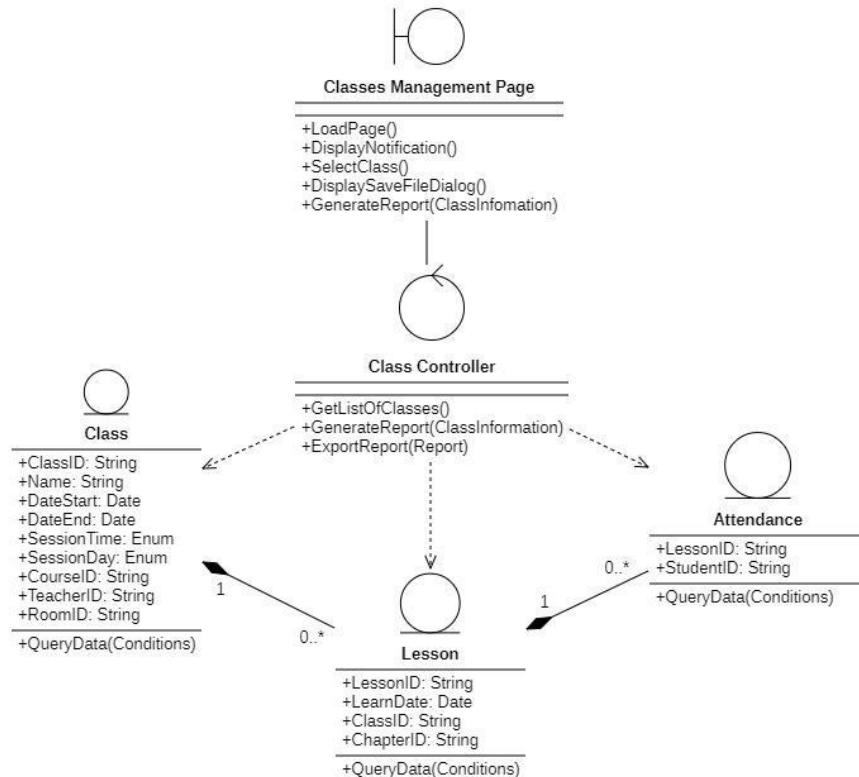


Figure 3. 75. Class Diagram for “Export Class Report”

3.3.11. Analyze

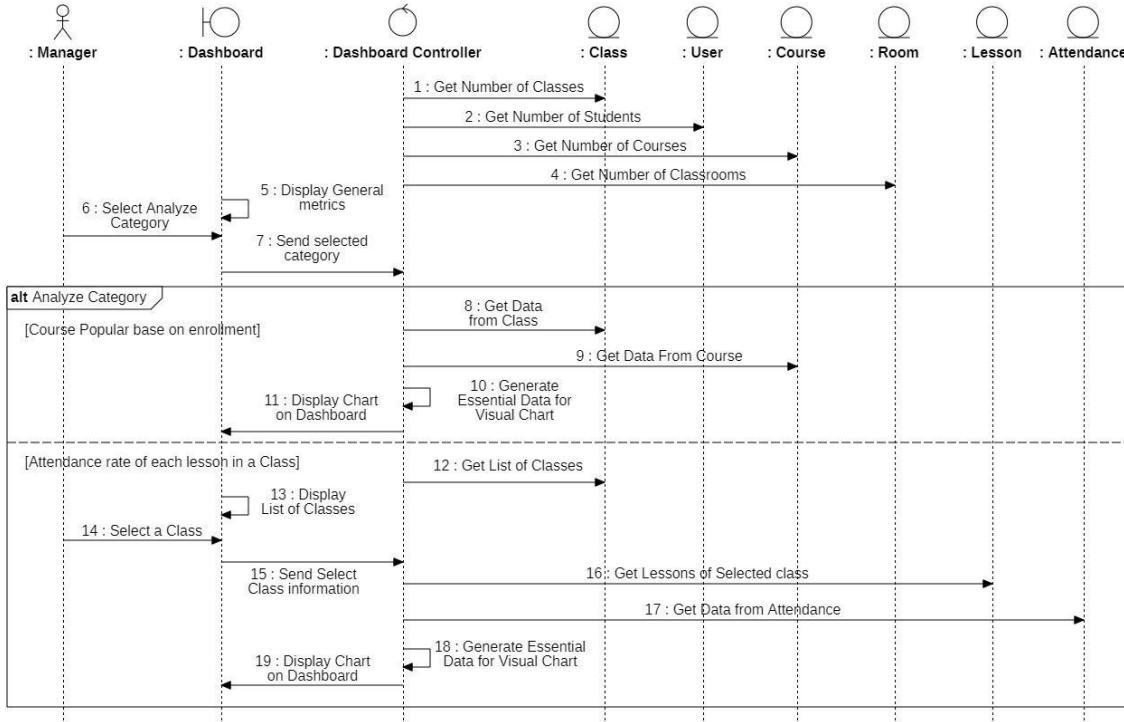


Figure 3. 76. Sequence Diagram for “Analyze”

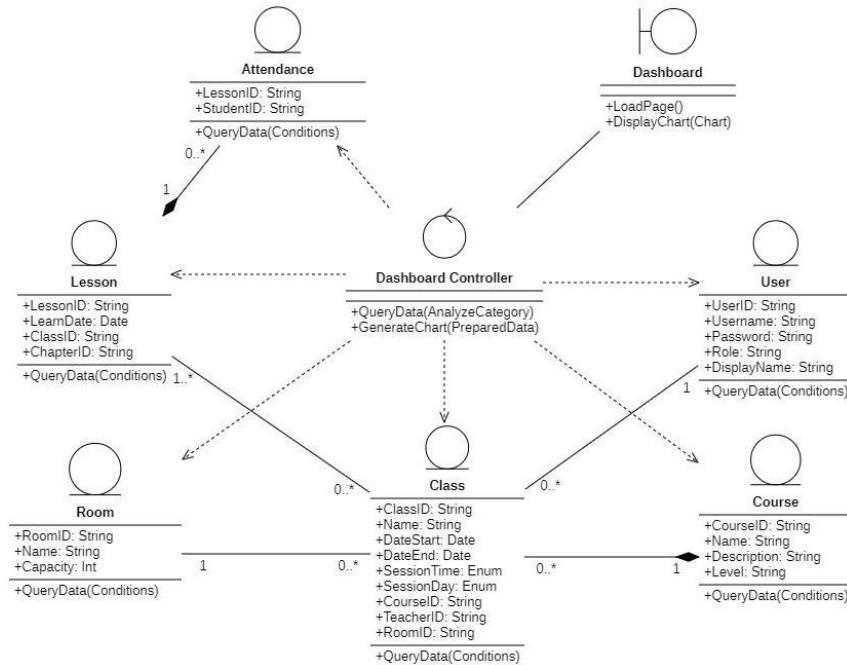


Figure 3. 77. Class Diagram for “Analyze”

3.3.12. Logout

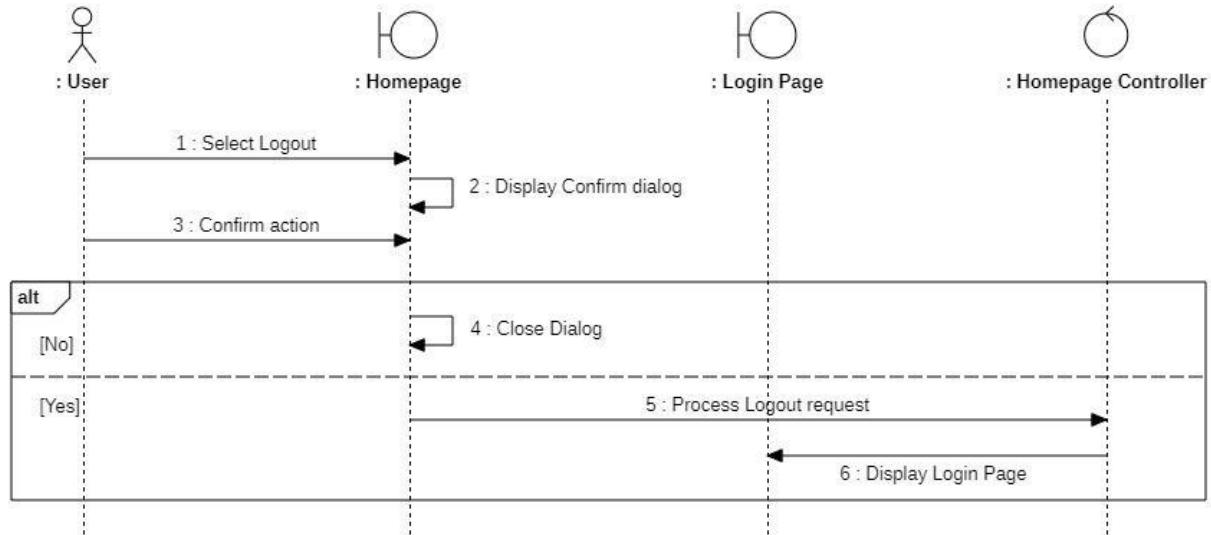


Figure 3. 78. Sequence Diagram for “Logout”

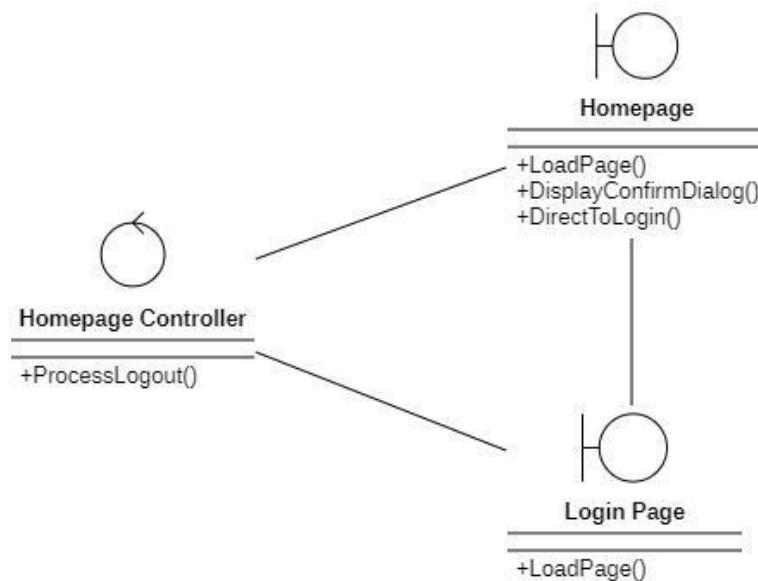


Figure 3. 79. Class Diagram for “Logout”

3.3.13. Explore Courses

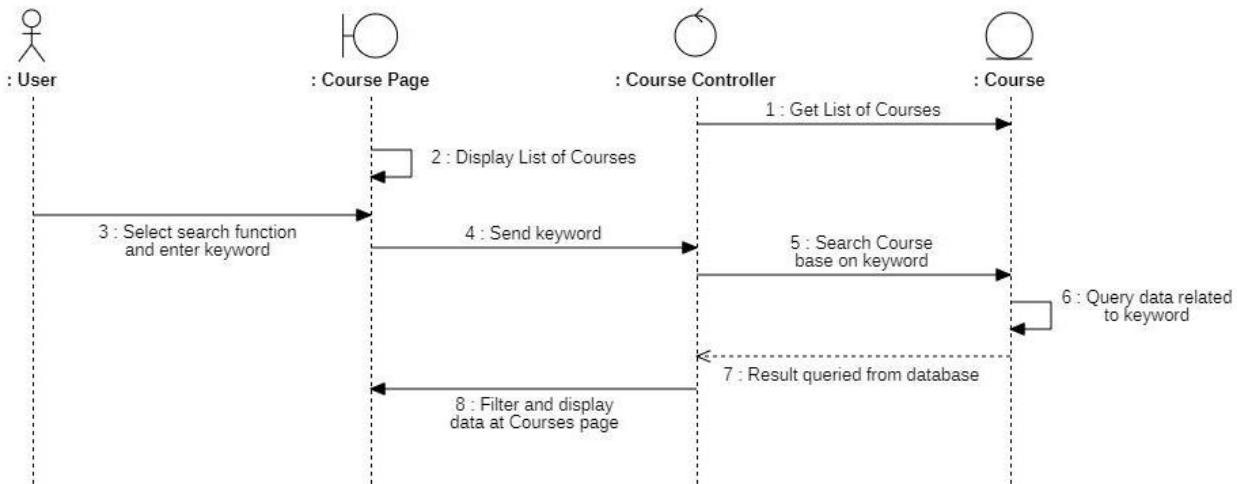


Figure 3. 80. Sequence Diagram for "Explore Courses"

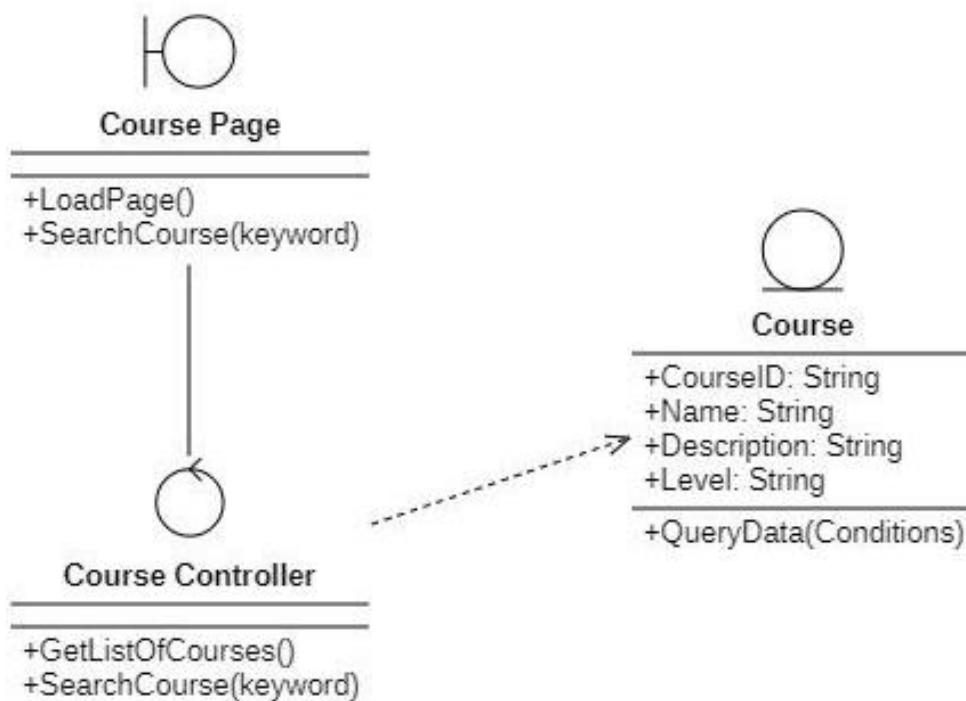


Figure 3. 81. Class Diagram for "Explore Courses"

3.3.14. Explore Chapters

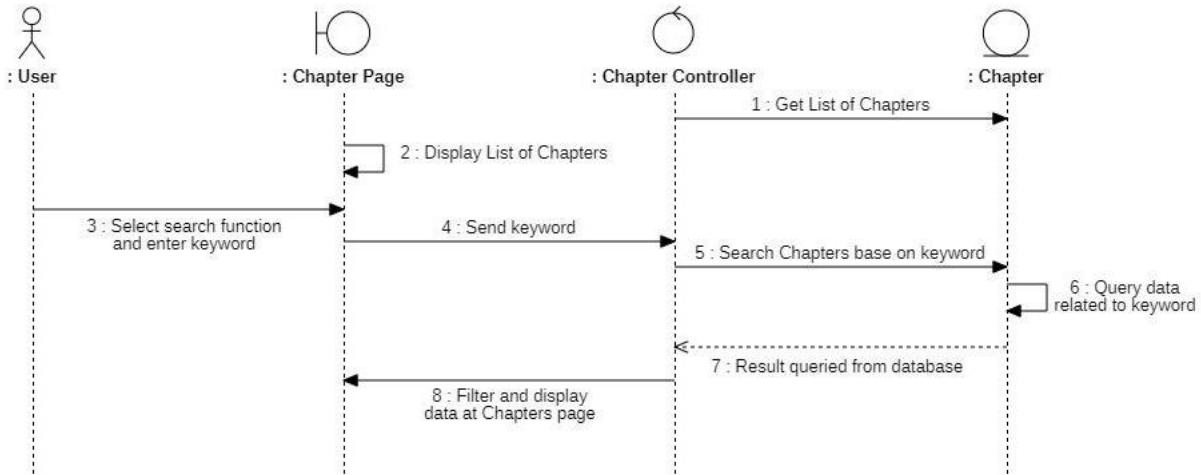


Figure 3. 82. Sequence Diagram for “Explore Chapters”

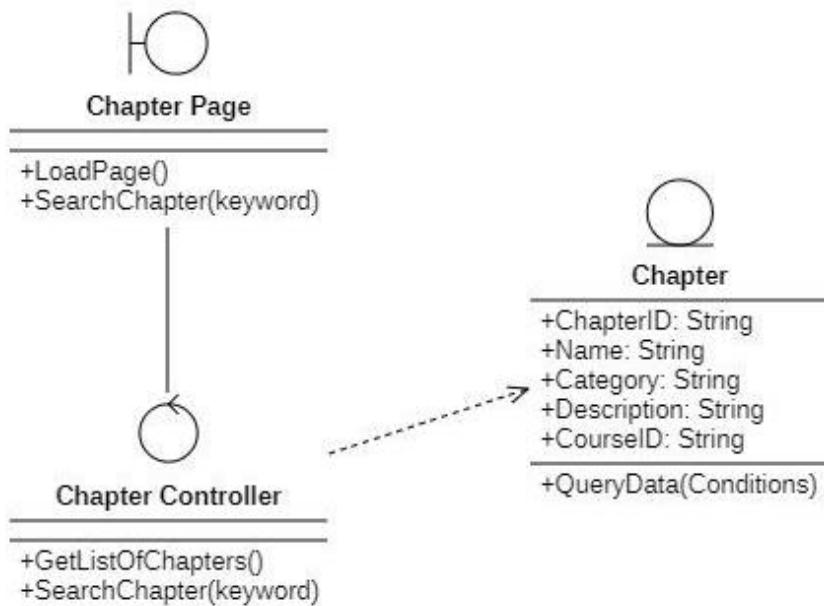


Figure 3. 83. Class Diagram for “Explore Chapters”

3.3.15. View Schedule

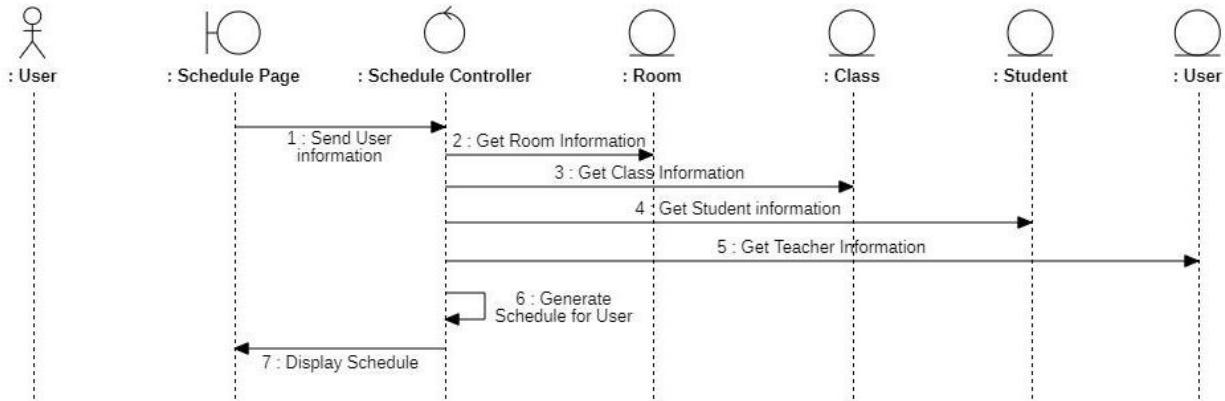


Figure 3. 84. Sequence Diagram for “View Schedule”

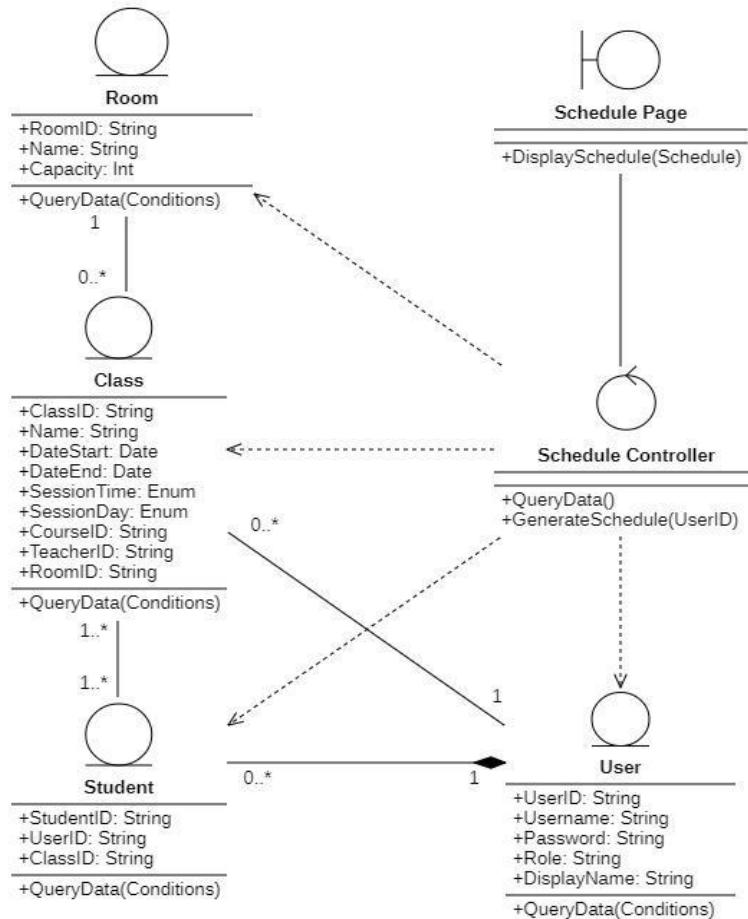


Figure 3. 85. Class Diagram for “View Schedule”

3.3.16. Manage Classes (for Teacher)

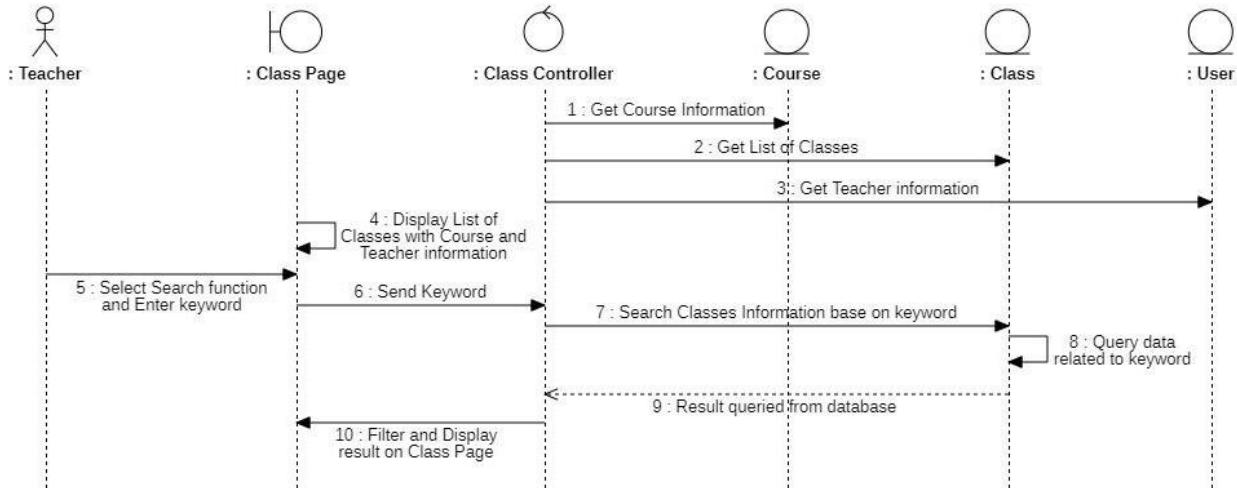


Figure 3. 86. Sequence Diagram for “Manage Classes (for Teacher)”

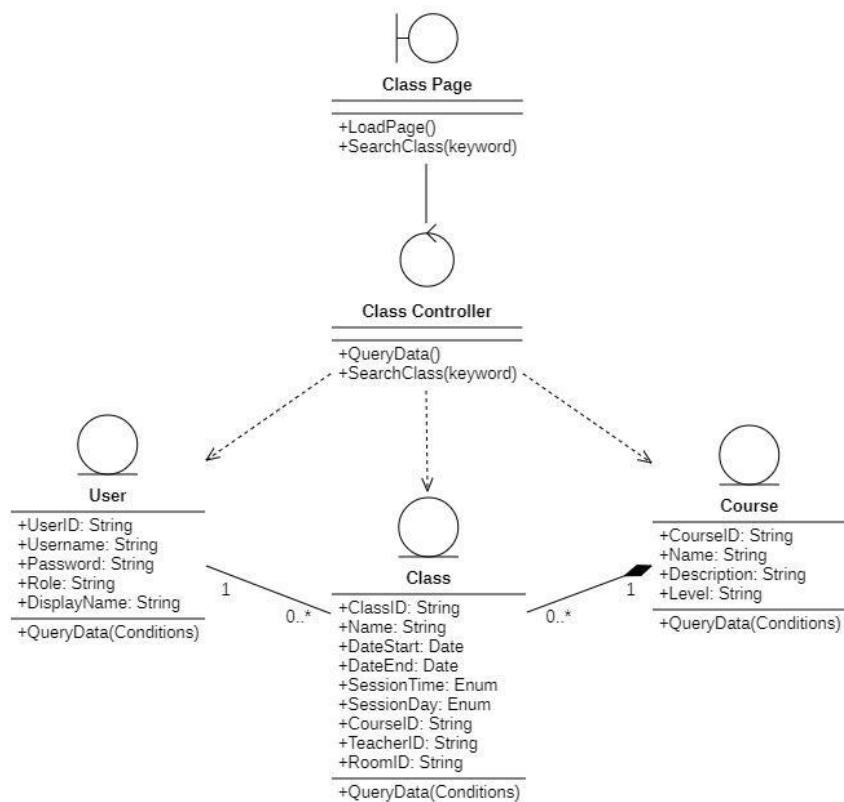


Figure 3. 87. Class Diagram for “Manage Classes (for Teacher)”

3.3.17. Manage Lessons (for Teacher)

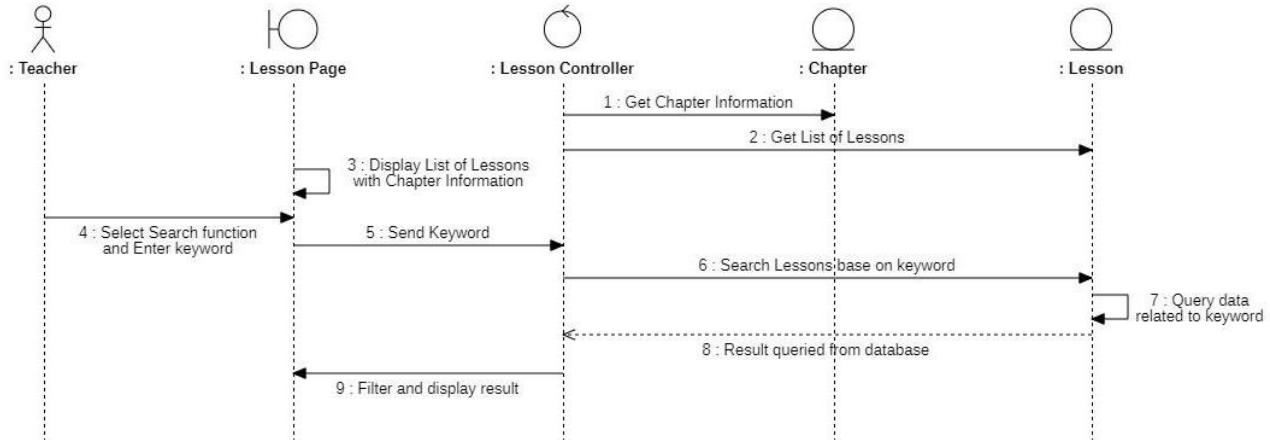


Figure 3. 88. Sequence Diagram for “Manage Lessons (for Teacher)”

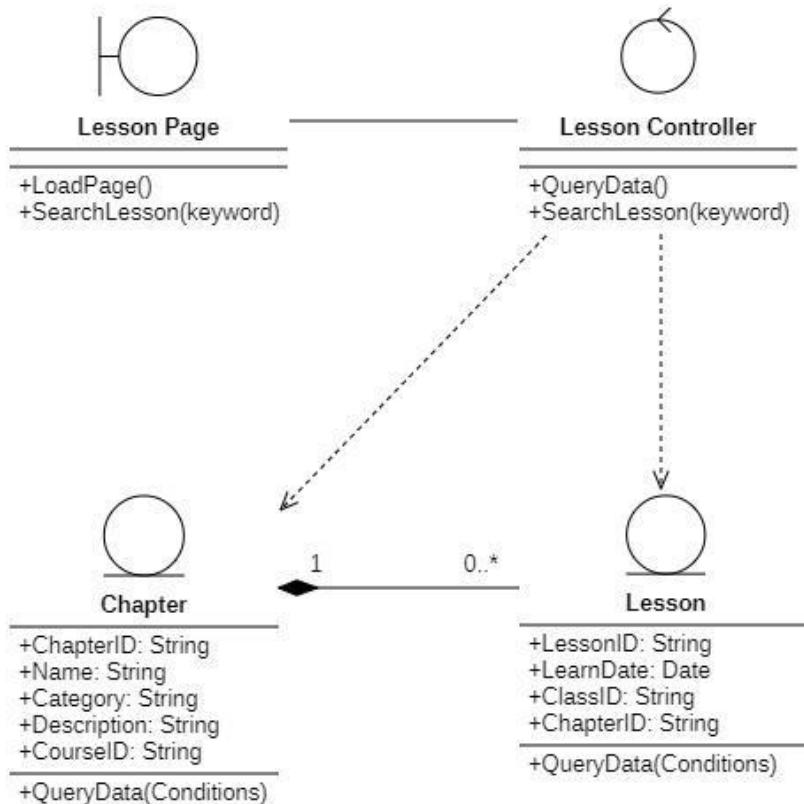


Figure 3. 89. Class Diagram for “Manage Lessons (for Teacher)”

3.3.18. View Classrooms

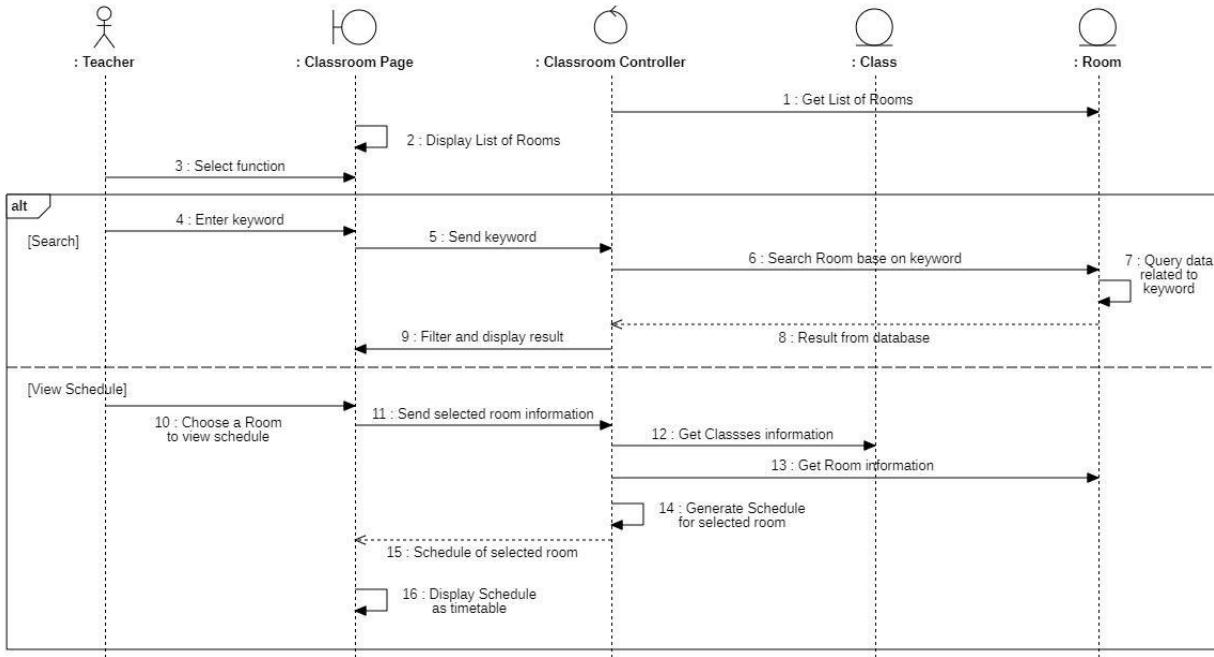


Figure 3. 90. Sequence Diagram for “View Classrooms”

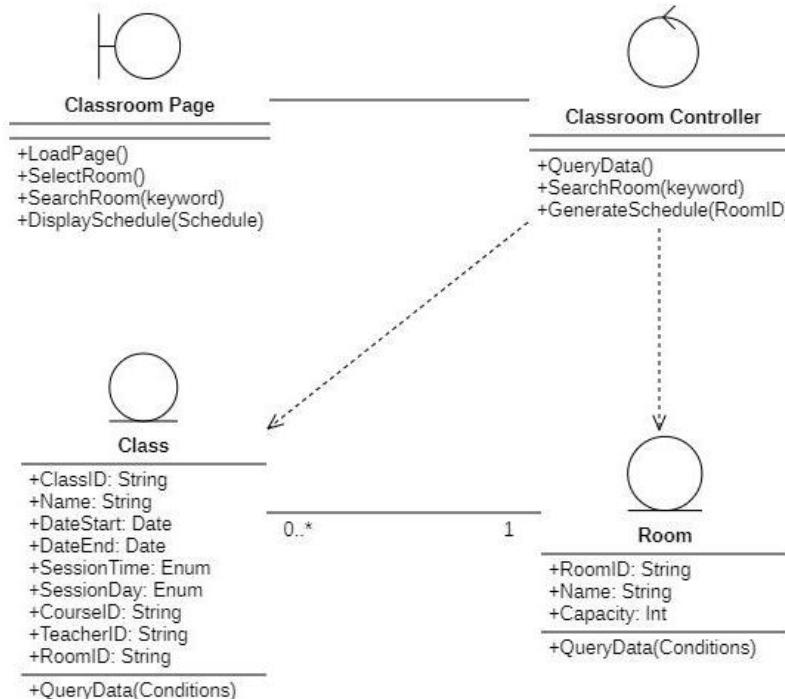


Figure 3. 91. Class Diagram for “View Classrooms”

3.3.19. View Classes

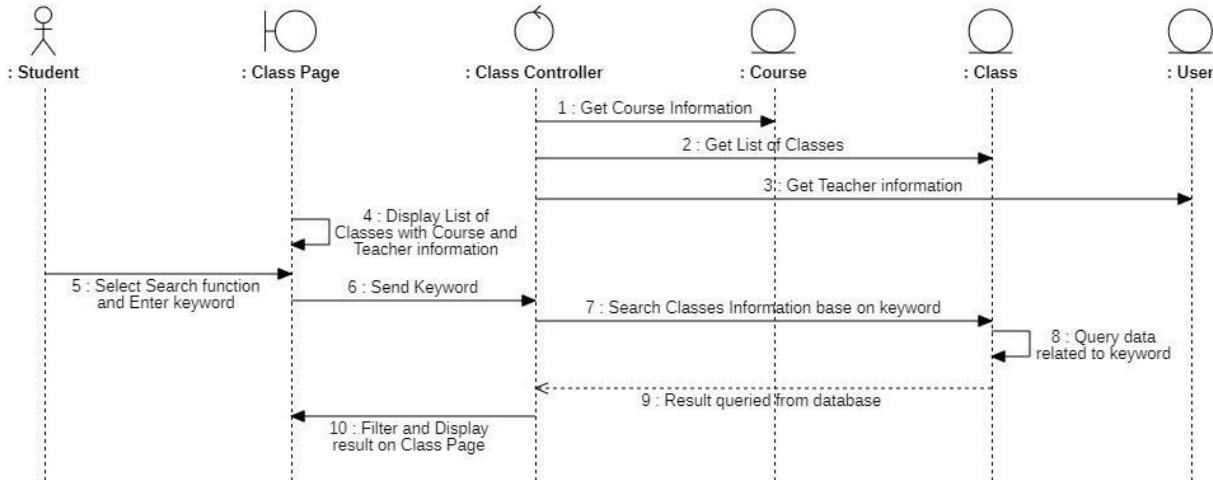


Figure 3. 92. Sequence Diagram for “View Classes”

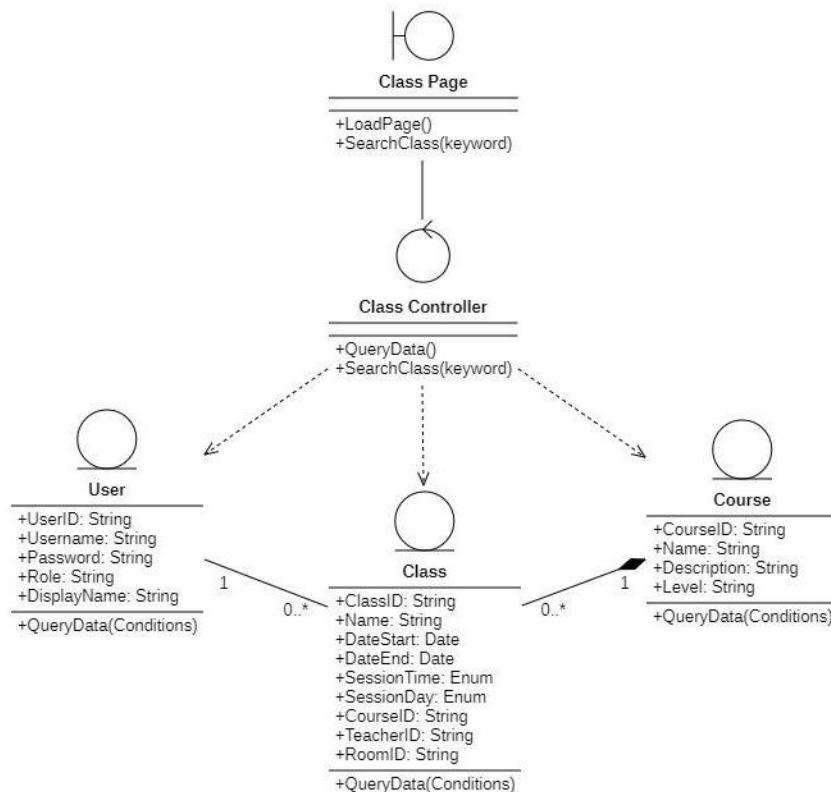


Figure 3. 93. Class Diagram for “View Classes”

3.3.20. View Lessons

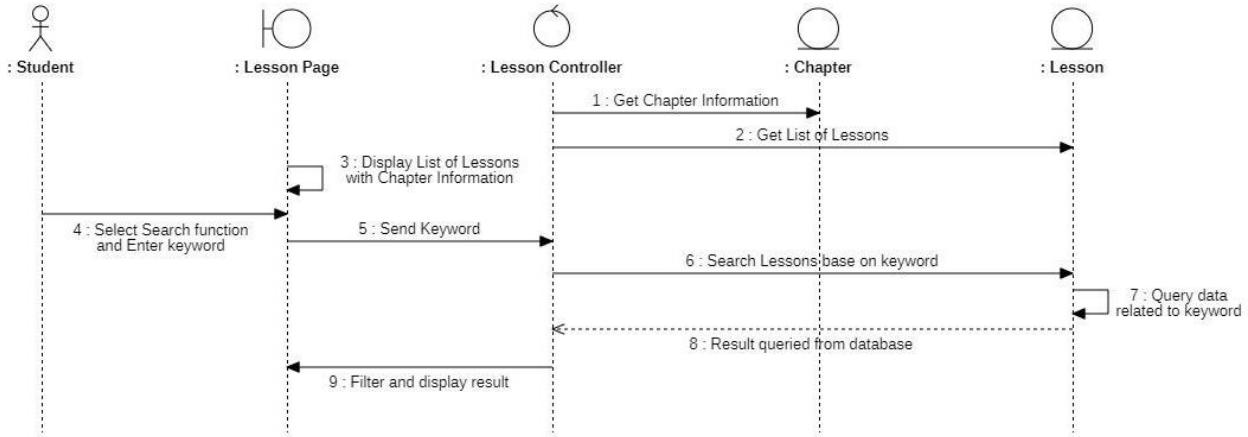


Figure 3. 94. Sequence Diagram for “View Lessons”

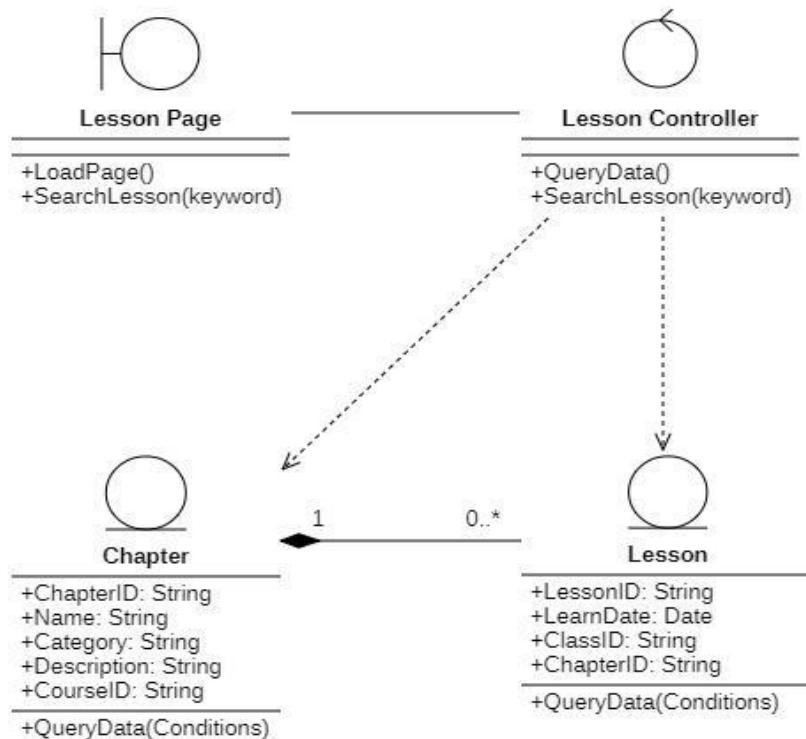


Figure 3. 95. Class Diagram for “View Lessons”

3.3.21. View Materials

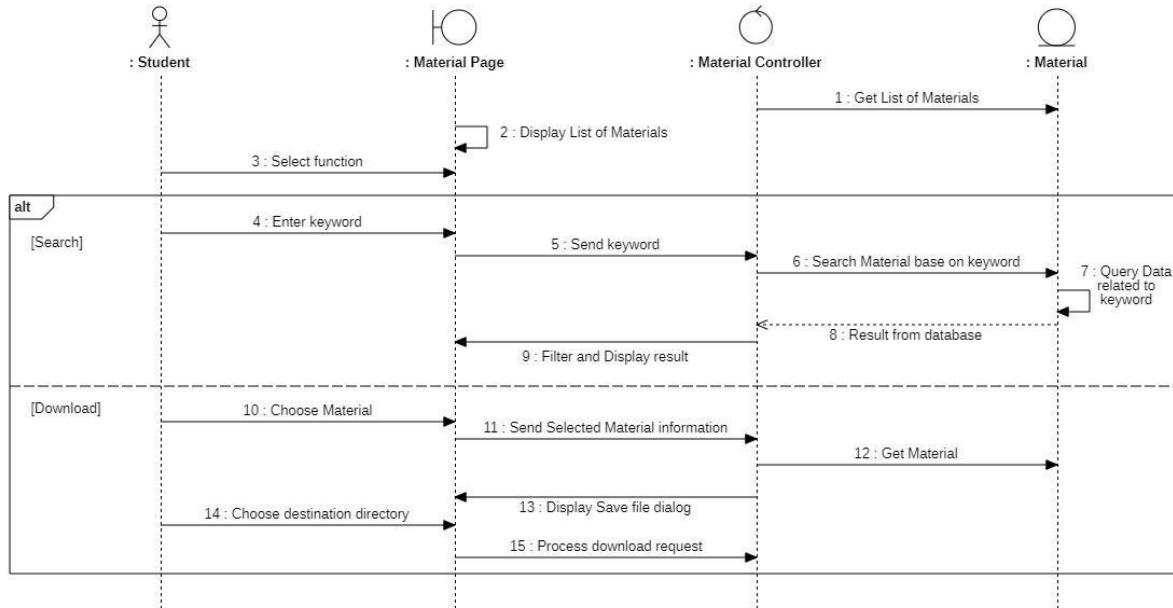


Figure 3. 96. Sequence Diagram for “View Materials”

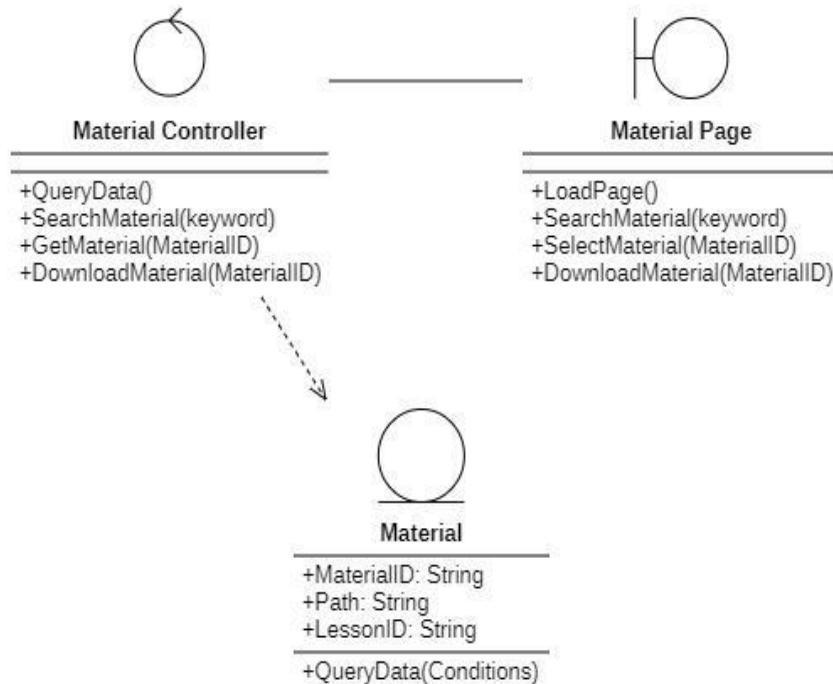


Figure 3. 97. Class Diagram for “View Materials”

3.4. Class Diagram and Entity class diagram

3.4.1. Overview Class diagram

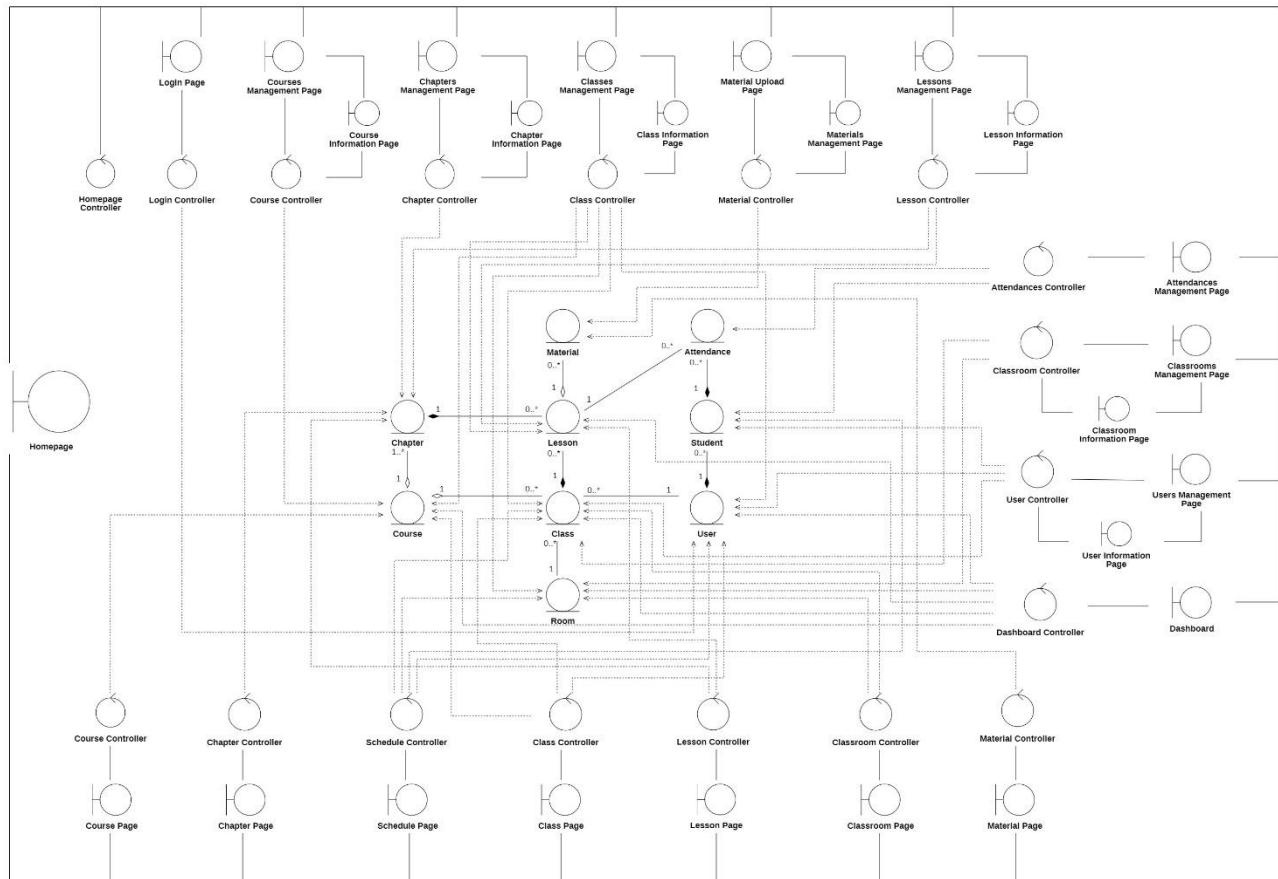


Figure 3. 98. Overview Class Diagram

3.4.2. Entity Class diagram

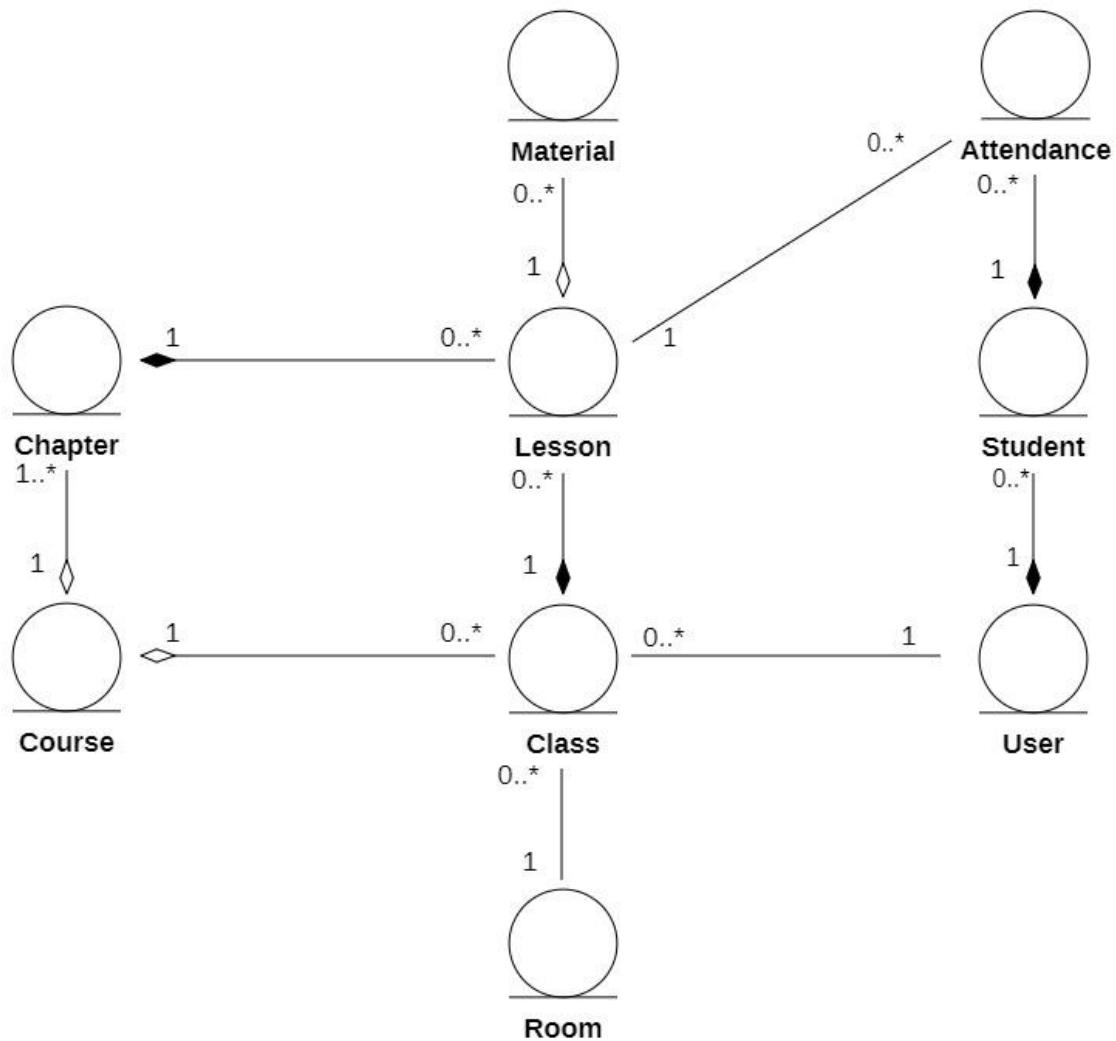


Figure 3. 99. Entity Class Diagram

3.4.3. Class diagram specification

3.4.3.1. Course

Course		
Attribute	Course_ID: String	Course ID generated by system automatically
	Name: String	Name of the Course
	Description: String	Description of the Course
	Course_level: String	Level of the Course base on difficulty
Operation	getNewID(): String	Return new ID for the new Course, unique in database
	QueryData(Conditions): List	Return the result of the query from database base on given conditions
	StoreData(CourseInfo): Boolean	<ul style="list-style-type: none"> - Insert a new record based on the new Course information if there is no existing version of that Course yet. - Update the course information if there is already an existing Course in the database. <p>Then return the result of that section (Boolean)</p>

3.4.3.2. Chapter

Chapter		
Attribute	Chapter_ID: String	Chapter ID generated by system automatically
	Name: String	Name of the Chapter
	Description: String	Description of the Chapter
	Category: String	Classify the chapters according to specific skills
	Course_ID: String	The ID of the Course contains this Chapter
Operation	getNewID(): String	Return new ID for the new Chapter, unique in database
	QueryData(Conditions): List	Return the result of the query from database base on given conditions
	StoreData(ChapterInfo): Boolean	<ul style="list-style-type: none"> - Insert a new record based on the new Chapter information if there is no existing version of that Chapter yet. - Update the Chapter information if there is already an existing version in the database.

Chapter		
		Then return the result of that section (Boolean)
	DeleteData(ChapterID): Boolean	Delete the Chapter in database base on given ChapterID. Then return the result of that section (Boolean)

3.4.3.3. Class

Class		
Attribute		
	Class_ID: String	Class ID generated by system automatically
	Name: String	Name of the Class
	DateStart: Date	The Start date of the Class
	DateEnd: Date	The End date of the Class
	SessionTime: Enum	Session Time for schedule of the Class
	SessionDay: Enum	Session Day for schedule of the Class
	Course_ID: String	The ID of the course that this class is currently teaching

Class		
Operation	Teacher_ID: String	The ID of the teacher who is teaching this class.
	Room_ID: String	The ID of the room where this class is currently being held.
	getNewID(): String	Return new ID for the new Class, unique in database
	QueryData(Conditions): List	Return the result of the query from database base on given conditions
	StoreData(ClassInfo): Boolean	<ul style="list-style-type: none"> - Insert a new record based on the new Class information if there is no existing version of that Class yet. - Update the Class information if there is already an existing version in the database. <p>Then return the result of that section (Boolean)</p>
	DeleteData(ClassID): Boolean	Delete the Chapter in database base on given ChapterID. Then return the result of that section (Boolean)

3.4.3.4. Lesson

Lesson		
Attribute		
Attribute	Lesson_ID: String	Lesson_ID generated by system automatically
	LearnDate: Date	The date which the class session is scheduled to teach.
	Class_ID: String	The ID of the class organizing this lesson
	Chapter_ID: String	The ID of the Chapter that this lesson will teach
Operation	getNewID(): String	Return new ID for the new Lesson, unique in database
	QueryData(Conditions): List	Return the result of the query from database base on given conditions
	StoreData(LessonInfo): Boolean	<ul style="list-style-type: none"> - Insert a new record based on the new Lesson information if there is no existing version of that Lesson yet. - Update the Lesson information if there is already an existing version in the database. <p>Then return the result of that</p>

Lesson		
		section (Boolean)
	DeleteData(LessonID): Boolean	Delete the Lesson in database base on given LessonID. Then return the result of that section (Boolean)

3.4.3.5. Material

Material		
Attribute	Material_ID: String	Material_ID generated by system automatically
	Path: String	The file path of the material on the computer.
	Lesson_ID: String	The ID of the class that contains this material.
Operation	QueryData(Conditions): List	Return the result of the query from database base on given conditions
	StoreData(MaterialInfo): Boolean	<ul style="list-style-type: none"> - Insert a new record based on the new Material information if there is no existing version of that Material yet. - Update the Material information

Material	
	if there is already an existing version in the database. Then return the result of that section (Boolean)
DeleteData(MaterialID): Boolean	Delete the Material in database base on given MaterialID. Then return the result of that section (Boolean)

3.4.3.6. Attendance

Attendance		
Attribute	Lesson_ID: String	The ID of the Class in which this attendance is recorded
	Student_ID: String	The ID of the Student which this attendance is recorded
	QueryData(Conditions): List	Return the result of the query from database base on given conditions
Operation	StoreData(AttendanceInfo): Boolean	- Insert a new record based on the new Attendance information if there is no existing version of that Attendance yet. - Update the Attendance

Attendance		
		information if there is already an existing version in the database. Then return the result of that section (Boolean)
	DeleteData(AttendanceID): Boolean	Delete the Attendance in database base on given AttendanceID. Then return the result of that section (Boolean)

3.4.3.7. Room

Room		
Attribute		
	Room_ID: String	Room_ID generated by system automatically
	Name: String	Name of the classroom
	Capacity: Integer	The maximum capacity of the classroom
Operation		
	getNewID(): String	Return new ID for the new Classroom, unique in database
	QueryData(Conditions): List	Return the result of the query from database base on given conditions

Room		
	StoreData(classroomInfo): Boolean	<ul style="list-style-type: none"> - Insert a new record based on the new classroom information if there is no existing version of that classroom yet. - Update the classroom information if there is already an existing version in the database. <p>Then return the result of that section (Boolean)</p>
	DeleteData(classroomID): Boolean	Delete the classroom in database base on given classroomID. Then return the result of that section (Boolean)

3.4.3.8. Student

Student		
Attribute		
	Student_ID: String	Student_ID generated by system automatically
	User_ID: String	The ID of the student as a user in system
	Class_ID: Integer	The ID of the class that this student is currently enroll

Student		
Operation	getNewID(): String	Return new ID for the new Student, unique in database
	QueryData(Conditions): List	Return the result of the query from database base on given conditions
	StoreData(StudentInfo): Boolean	<ul style="list-style-type: none"> - Insert a new record based on the new student information if there is no existing version of that student yet. - Update the student information if there is already an existing version in the database. <p>Then return the result of that section (Boolean)</p>
	DeleteData(StudentID): Boolean	Delete the student in database base on given studentID. Then return the result of that section (Boolean)

3.4.3.9. User

User		
Attribute	User_ID: String	Student_ID generated by system automatically
	Username: String	The account name of user use for login into system
	Password: Integer	The password of user use for login into system
	Role: String	The role of user in system
	DisplayName: String	The name of user that display in system interface
Operation	getNewID(): String	Return new ID for the new User, unique in database
	QueryData(Conditions): List	Return the result of the query from database base on given conditions
	StoreData(UserInfo): Boolean	<ul style="list-style-type: none"> - Insert a new record based on the new User information if there is no existing version of that User yet. - Update the User information if there is already an existing

User		
		version in the database. Then return the result of that section (Boolean)
	DeleteData(UserID): Boolean	Delete the User in database base on given UserID. Then return the result of that section (Boolean)
	ValidateUserCredentialinDB(UserCredentials): Boolean	Validate user credential in database for login action

3.4.3.10. Login Controller

Login Controller		
Operation		
	ValidateUserCredentials(UserCredential): Boolean	Request Database to check User credentials then return result of the validation

3.4.3.11. Course Controller

Course Controller		
Operation	Method Signature	Description
Operation	GetListOfCourses(): List	Request Database to get list of all existed Courses
	GetCourseInfo(CourseID): Course	Request database to get Course information base on given CourseID
	AddCourse(CourseInfo): Boolean	Request database to insert new Course with given Course information. Return validation result
	UpdateCourse(CourseInfo): Boolean	Request database to update Course with given Course information. Return validation result
	DeleteCourse(CourseID): Boolean	Request database to delete Course base on given CourseID. Return validation result
	SearchCourse(Keyword): List	Request database to get list of Courses related to the given Keyword

3.4.3.12. Chapter Controller

Chapter Controller		
Operation		
	GetListOfChapters(): List	Request Database to get list of all existed Chapters
	GetChapterInfo(ChapterID): Chapter	Request database to get Chapter information base on given ChapterID
	AddChapter(ChapterInfo): Boolean	Request database to insert new Chapter with given Chapter information. Return validation result
	UpdateChapter(ChapterInfo): Boolean	Request database to update Chapter with given Chapter information. Return validation result
	DeleteChapter(ChapterID): Boolean	Request database to delete Chapter base on given ChapterID. Return validation result
	SearchChapter(Keyword): List	Request database to get list of Chapters related to the given Keyword

3.4.3.13. Class Controller

Class Controller		
Operation		
	GetListOfClasses(): List	Request Database to get list of all existed Classes
	GetClassInfo(ChapterID): Class	Request database to get Class information base on given ClassID
	AddClass(ClassInfo): Boolean	Request database to insert new Class with given Class information. Return validation result
	UpdateClass(ClassInfo): Boolean	Request database to update Class with given Class information. Return validation result
	DeleteClass(ClassID): Boolean	Request database to delete Class base on given ClassID. Return validation result
	SearchClass(Keyword): List	Request database to get list of Classes related to the given Keyword
	GenerateReport(ClassInformation): Report	Request database to query data for report then generate report for specific class

Class Controller		
	ExportReport(Report): Boolean	Export pdf version of report then process save file to user's computer

3.4.3.14. Lesson Controller

Lesson Controller		
Operation		
	GetListOfLessons(): List	Request Database to get list of all existed Lessons
	GetLessonInfo(LessonID): Class	Request database to get Lesson information base on given LessonID
	AddLesson(LessonInfo): Boolean	Request database to insert new Lesson with given Lesson information. Return validation result
	UpdateLesson(LessonInfo): Boolean	Request database to update Lesson with given Lesson information. Return validation result
	DeleteLesson(LessonID): Boolean	Request database to delete Lesson base on given LessonID. Return validation result

Lesson Controller		
	SearchLesson(Keyword): List	Request database to get list of Lessons related to the given Keyword

3.4.3.15. Material Controller

Material Controller		
Operation		
	GetListOfMaterials(): List	Request Database to get list of all existed Materials
	GetMaterial(MaterialID): Material	Request database to get Material information base on given MaterialID
	AddMaterial(MaterialInfo): Boolean	Request database to insert new Material with given Material information. Return validation result
	DownloadMaterial(MaterialInfo): Boolean	Base on Material information, process download Material for user
	DeleteMaterial(MaterialID): Boolean	Request database to delete Material base on given MaterialID. Return validation result

Material Controller		
	SearchMaterial(Keyword): List	Request database to get list of Materials related to the given Keyword
	ValidateFile(UploadFile): Boolean	Validate file upload from user, return result of validation

3.4.3.16. Attendance Controller

Attendance Controller		
Operation		
	GetListOfAttendances(): List	Request Database to get list of all existed Attendances
	GetAttendanceInfo(AttendanceID) : Class	Request database to get Attendance information base on given AttendanceID
	AddAttendance(AttendanceInfo): Boolean	Request database to insert new Attendance with given Attendance information. Return validation result
	UpdateAttendance(AttendanceInfo): Boolean	Request database to update Attendance with given Attendance information. Return validation result

Attendance Controller		
	DeleteAttendance(AttendanceID): Boolean	Request database to delete Attendance base on given AttendanceID. Return validation result
	SearchAttendance(Keyword): List	Request database to get list of Attendances related to the given Keyword

3.4.3.17. Classroom Controller

Classroom Controller		
Operation	GetListOfClassrooms(): List	Request Database to get list of all existed Classrooms
	GetClassroomInfo(MaterialID): Class	Request database to get Classroom information base on given ClassroomID
	AddClassroom(ClassroomInfo): Boolean	Request database to insert new Classroom with given Classroom information. Return validation result
	UpdateClassroom(ClassroomInfo) : Boolean	Request database to update Classroom with given Classroom information. Return

Classroom Controller		
		validation result
	DeleteClassroom(ClassroomID): Boolean	Request database to delete Classroom base on given ClassroomID. Return validation result
	SearchClassroom(Keyword): List	Request database to get list of Classrooms related to the given Keyword
	GenerateSchedule(ClassroomID)	Request data from database base on ClassroomID then generate Schedule for this room

3.4.3.18. User Controller

User Controller		
Operation		
	GetListOfUsers(): List	Request Database to get list of all existed Users
	GetUserInfo(UserID): Class	Request database to get User information base on given UserID
	AddUser(UserInfo): Boolean	Request database to insert new User with given User

User Controller		
		information. Return validation result
	UpdateUser(UserInfo): Boolean	Request database to update User with given User information. Return validation result
	DeleteUser(UserID): Boolean	Request database to delete User base on given UserID. Return validation result
	SearchUser(Keyword): List	Request database to get list of Users related to the given Keyword
	EnrollStudent(UserID, ClassID)	Process Class Enrollment for selected user

3.4.3.19. Dashboard Controller

Dashboard Controller		
Operation	QueryData(AnalyzeCategory): List	Request Database to prepare data for selected category
	GenerateChart(PreparedData): Chart	Generate Chart for prepared data then return a visual Chart

3.4.3.20. Schedule Controller

Schedule Controller		
Operation	QueryData(): List	Request Database to prepare data for schedule
	GenerateSchedule(UserID): Schedule	Generate schedule for User with data queried from database

3.4.3.21. Homepage Controller

Schedule Controller		
Operation	ProcessLogout(): Void	Process Logout request for user

3.5. State Diagram

3.5.1. User

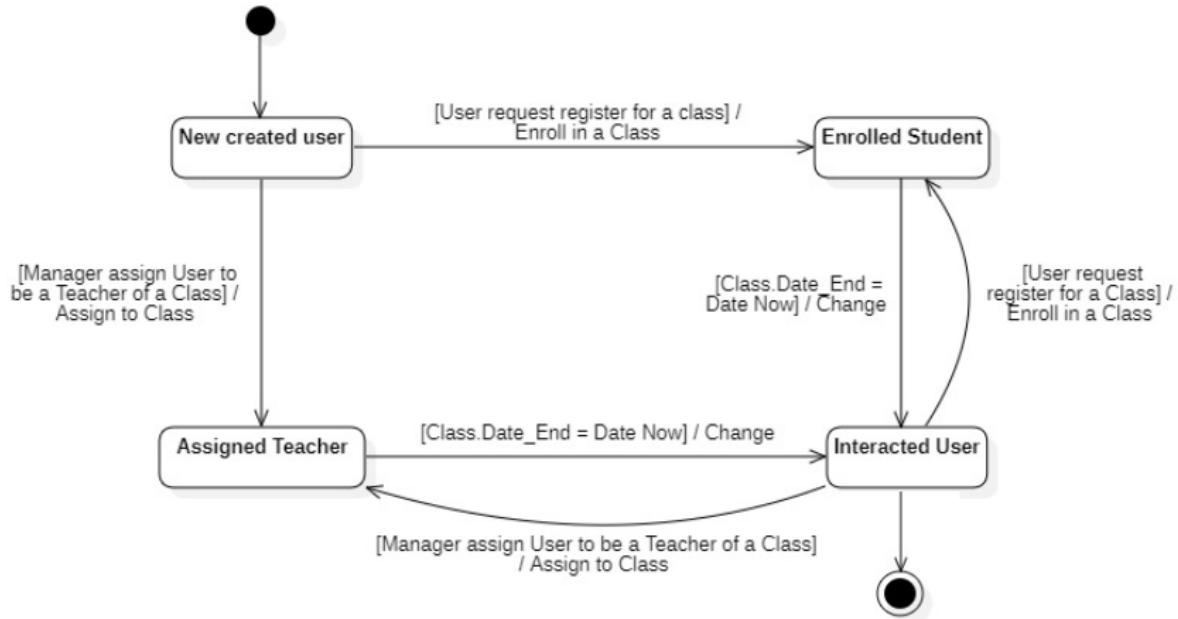


Figure 3. 100. State Diagram for **User**

3.5.2. Class

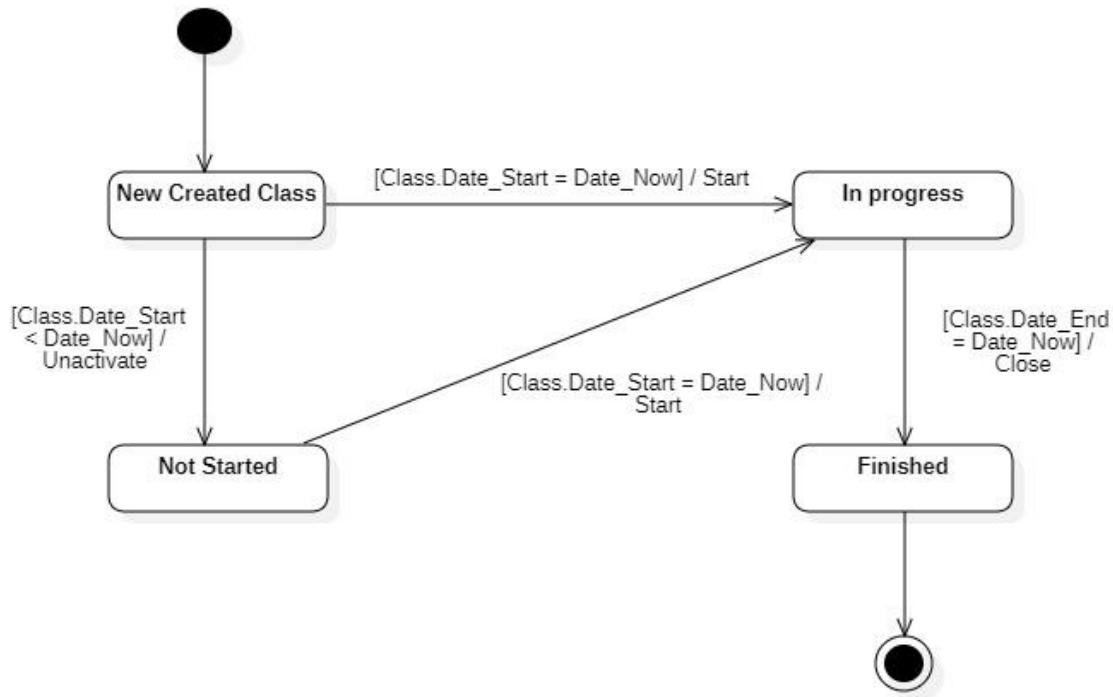


Figure 3. 101. State Diagram for **Class**

3.5.3. Classroom

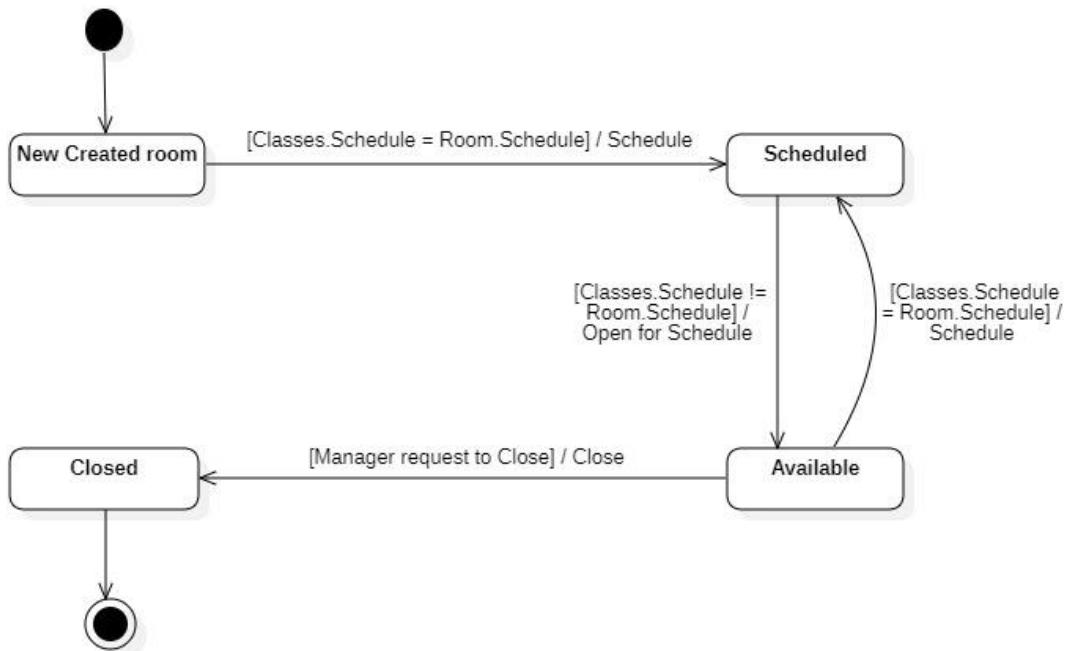


Figure 3. 102. State Diagram for **Classroom**

3.6. Mapping to Relational Model from Entity class diagram

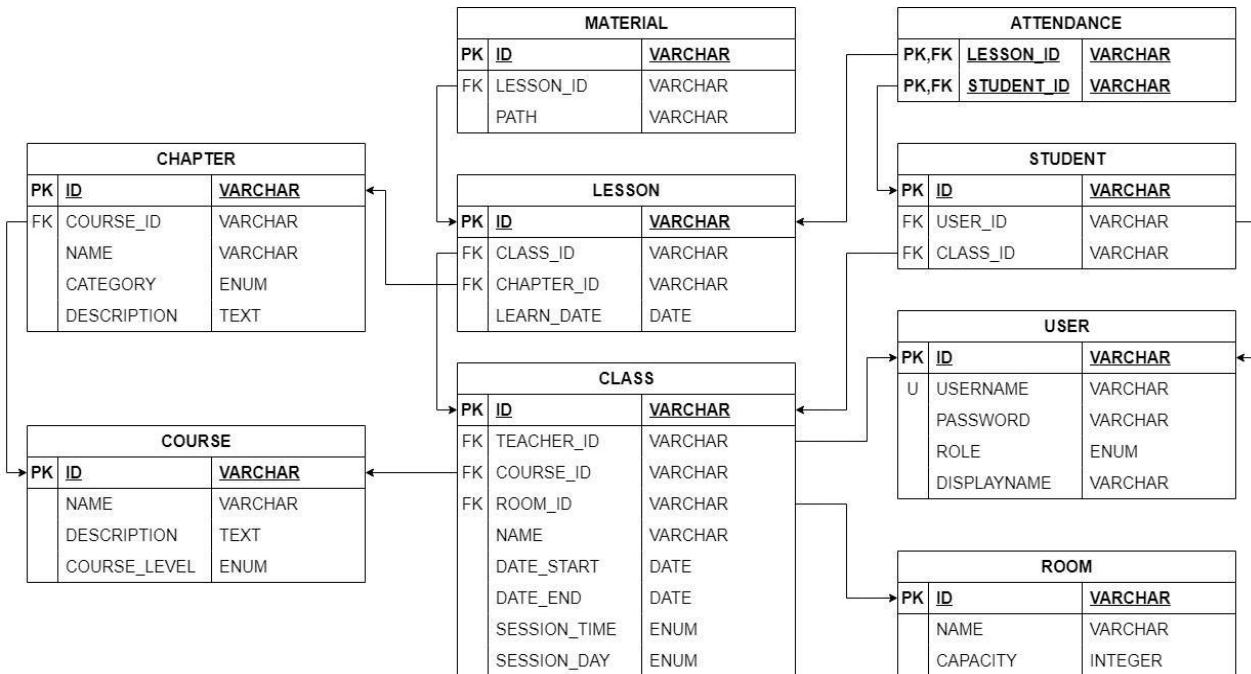


Figure 3. 103. Relational Model Diagram

Chapter 4: IMPLEMENTATION AND TESTING

4.1. Introduction to the implementation environment

- Operating System: Window 11 / MacOS Ventura
- Database: Oracle database
- IDE: IntelliJ IDEA
- Programming language: Java
- Library: JavaFX, Lombok, PDFBox,..

4.2. User Interface

4.2.1. Login

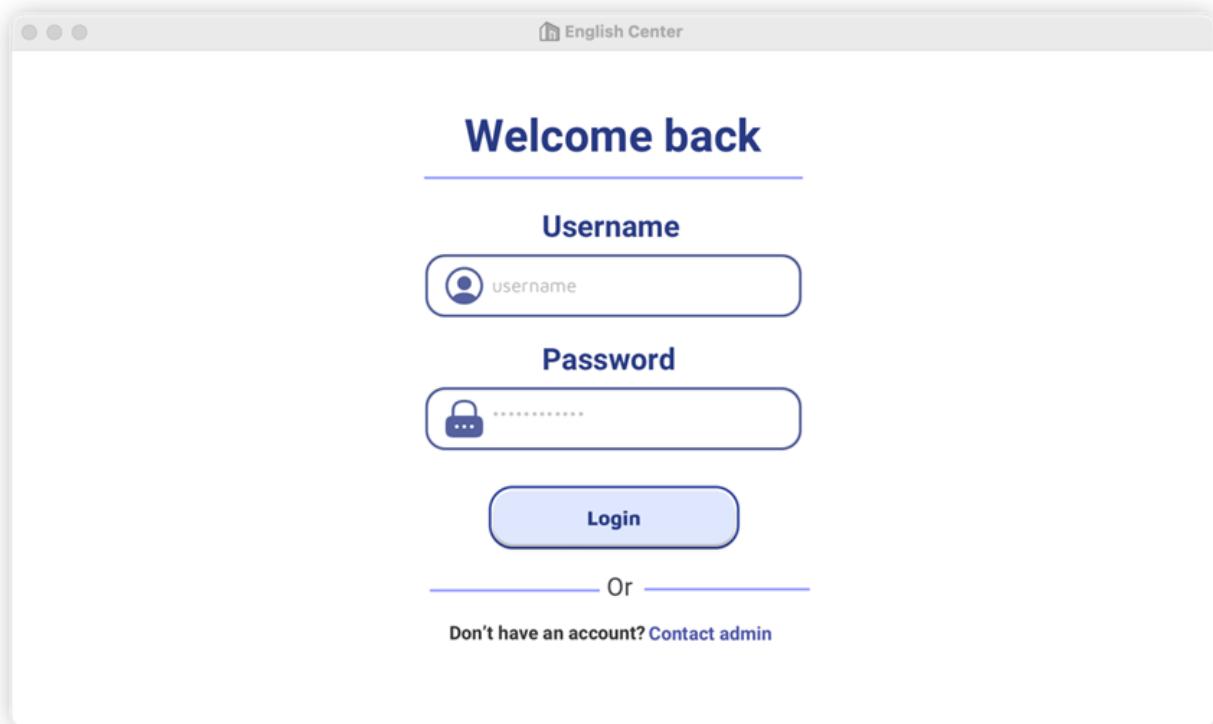


Figure 4. 1. Login page

4.2.2. Description about functions of manager

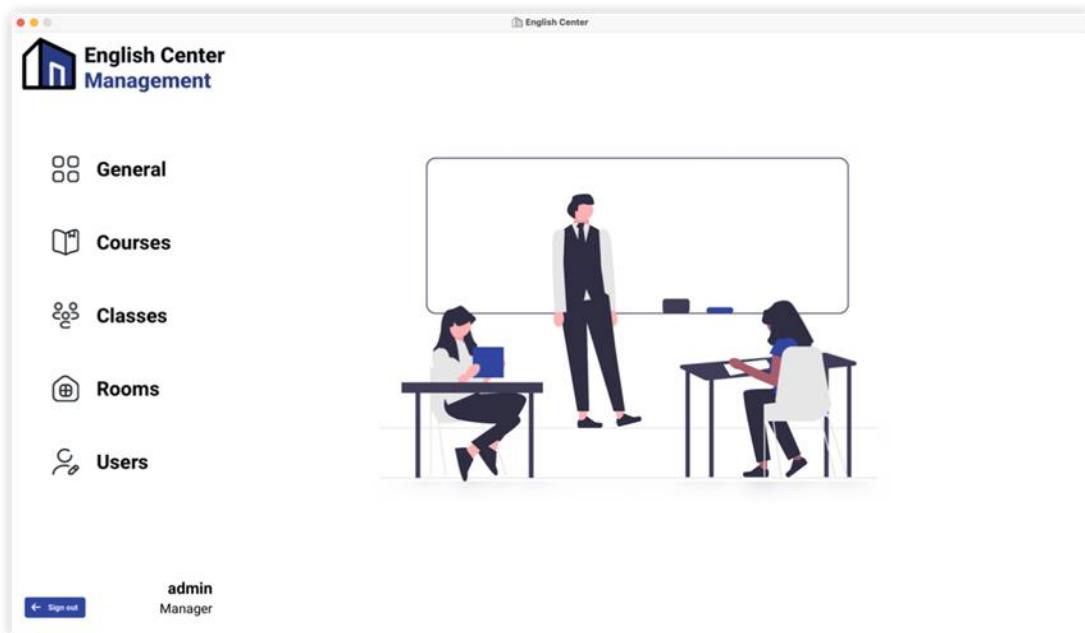


Figure 4. 2. Homepage

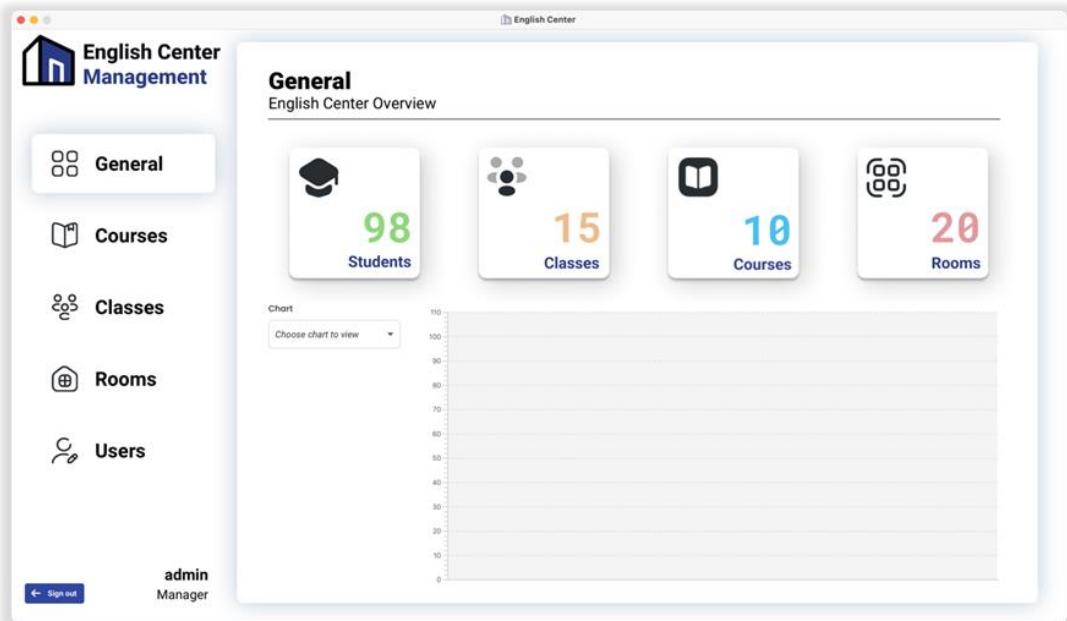


Figure 4. 3. General page

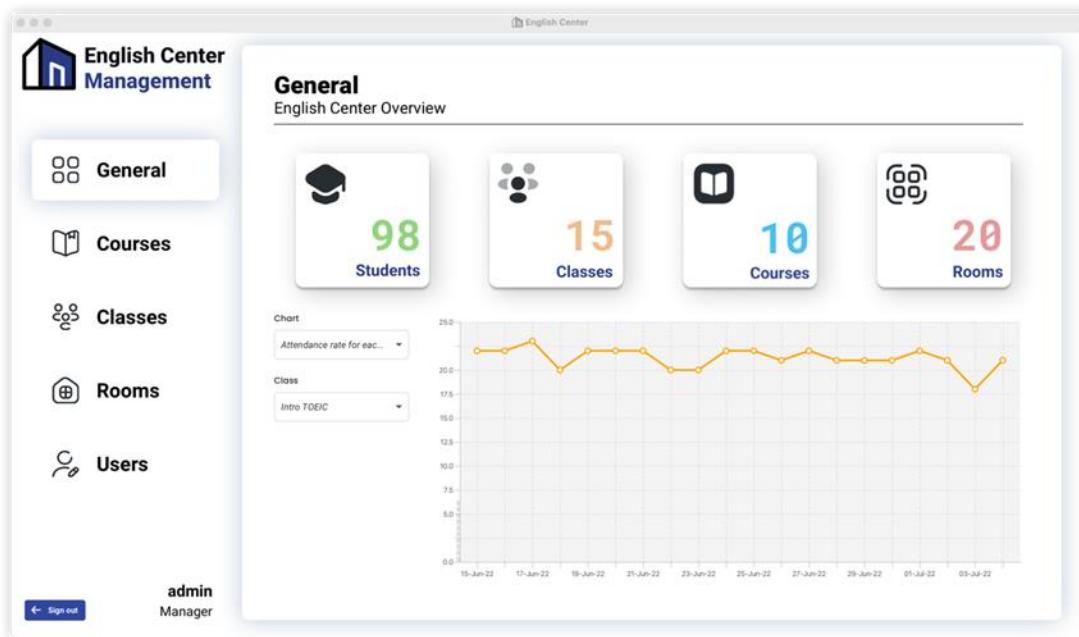


Figure 4.4. General page

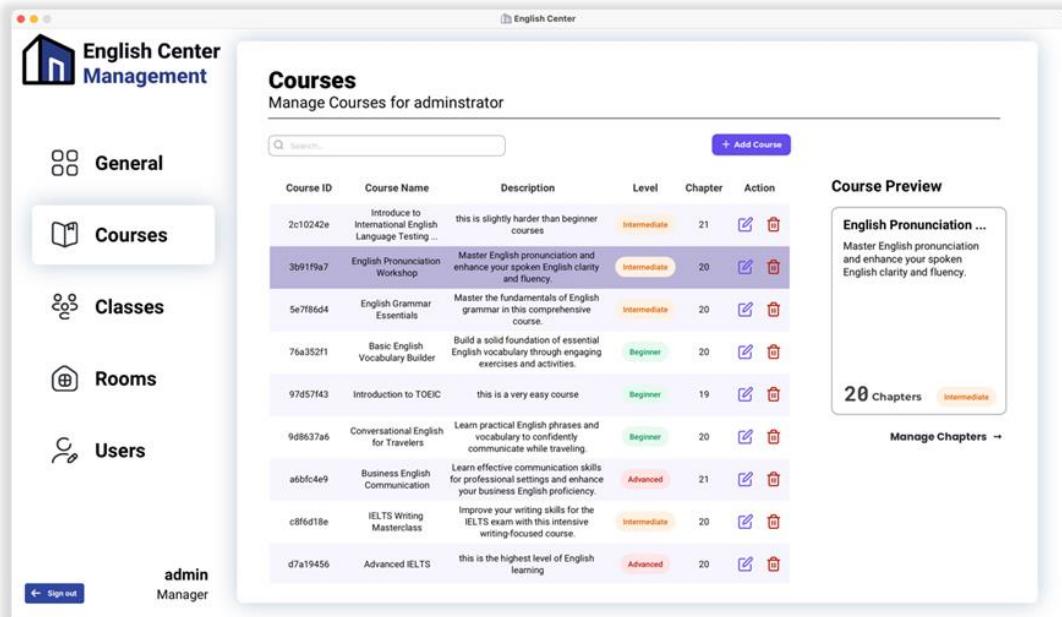


Figure 4.5. Courses page

X

Course Information

Fill in the data for course.

Course ID

3b91f9a7

Course Name

English Pronunciation Workshop

Description

Master English pronunciation and enhance your spoken English clarity and fluency.

Level

Intermediate ▾

Submit

Figure 4. 6. Course's information updating function UI

Course ID	Course Name	Description	Level	Chapter	Action	Course Preview
2c10242e	Introduce to International English Language Testing ...	this is slightly harder than beginner courses	Intermediate	21		Select course to preview
3b91f9a7	English Pronunciation Workshop	Master English pronunciation and enhance your spoken English clarity and fluency.	Intermediate	20		
5e7f86d4	English Grammar Essentials	Master the fundamentals of English grammar in this comprehensive course.	Intermediate	20		
76a352f1	Basic English Vocabulary Builder	Build a solid foundation of essential English vocabulary through engaging exercises and activities.	Beginner	20		
97d57f43	Introduction to TOEIC	this is a very easy course	Beginner	19		
9d8637a6	Conversational English for Travelers	Learn practical English phrases and vocabulary to confidently communicate while traveling.	Beginner	20		
a6bfc4e9	Business English Communication	Learn effective communication skills for professional settings and enhance your business English proficiency.	Advanced	21		
c8f6d18e	IELTS Writing Masterclass	Improve your writing skills for the IELTS exam with this intensive writing-focused course.	Intermediate	20		
d7a19456	Advanced IELTS	this is the highest level of English learning	Advanced	20		

Figure 4. 7. Chapters page

Chapter Information
Fill in the data for chapter.

Chapter ID	562d38d4
Chapter Name	Enter the chapter name
Description	Describe the chapter
Category	Choose chapter category

Submit

Figure 4. 8. Chapter information updating function UI

Class ID	Class Name	Course Name	Teacher	Room	Start date	Finish Date	Action
7fc90be3	Intro TOEIC	Introduction to TOEIC	Nguyen Thi Thuy	ENG1	15-Jun-22	15-Jul-22	
ee8ff35c	Intro IELTS	Introduce to International English Language Testing System	Ninh Thien Bao	ENG2	22-Dec-22	22-Jan-23	
c9f9e62f	Adv IELTS	Advanced IELTS	Tieu Quang Minh	ENG3	10-Sep-23	10-Oct-23	
a2f8db10	Eng Grammar	English Grammar Essentials	Le Anh Tuan	ENG4	05-Mar-22	05-Apr-22	
47d23b3a	Bus Eng Comm	Business English Communication	Nguyen Giang	ENG5	18-Jul-23	18-Aug-23	
ebe5e33e	TOEFL Prep	TOEFL Preparation Course	Nguyen Trung	ENG6	27-Nov-22	27-Dec-22	
c812936b	Conv Eng	Conversational English for Travelers	Tran Linh	ENG7	04-Feb-23	06-Mar-23	
3fd22f17	IELTS Writing	IELTS Writing Masterclass	Nguyen Thi Thuy	ENG8	12-Apr-22	12-May-22	

admin Manager [Sign out](#)

Figure 4. 9. Classes page

Class Information

Fill in the data for class.

Class ID	Session	Study Day
<input type="text" value="9622e2fe"/>	<input type="button" value="Choose session"/>	<input type="button" value="Choose day"/>
Name	Start date	Finish date
<input type="text" value="Enter the chapter name"/>	<input type="button" value="Choose date"/>	<input type="button" value="Choose date"/>
Course	Room	
<input type="button" value="Choose course"/>	<input type="button" value="Choose room"/>	
Teacher	<input type="button" value="Submit"/>	
<input type="button" value="Choose teacher"/>		

Figure 4. 10. Class's information updating function UI

English Center Management

Lesson
Manage Lessons in Class

Intro TOEIC
Introduction to TOEIC

Lesson Type	Section	Date
Listening	Part 1: Photographs	15-Jun-22
Listening	Part 2: Question-Response	16-Jun-22
Listening	Part 3: Conversations	17-Jun-22
Listening	Part 4: Short Talks	18-Jun-22
Writing	Writing Section Overview	19-Jun-22
Writing	Part 5: Sentence Complet...	20-Jun-22

Materials **Attendances**

Search... Upload File

Materials: <path>\Documents\H4\materials_09\7319a09d\listening part1.pdf

Back to Classes

Figure 4. 11. Lesson page

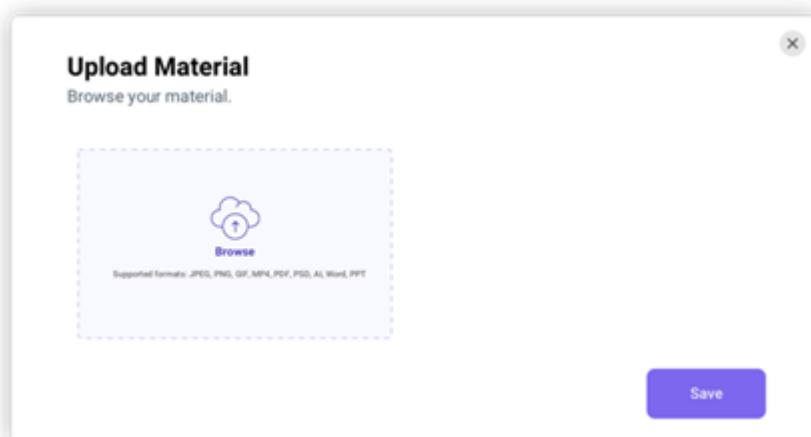


Figure 4. 12. Uploading materials function UI

Lesson
Manage Lessons in Class

Intro TOEIC
Introduction to TOEIC

Materials [Attendances](#)

Absent	Total
19	25

List of absent students

- 7baeaa1 Nguyen Minh Tuan
- 85e3423c Nguyen Hoang Nam
- 8da75b66 Hoang Nhat
- 9ef1aae Dao Tien Dat
- 9a7151ee Tuan Anh
- a47cf543 Thuy Hong
- ac63941 Tran Thuy Trang
- ad4726d0 Thanh Truc
- b7b2075a Tran Thuy Trang
- bc27df64 Nguyen Minh Trang
- bd9b363c Nguyen Tien Thanh
- d91b5e98 Nguyen Hoang Nam
- d97dd205 Nguyen Tuan Anh
- da0a0440 Quang Minh
- e9b91ec0 Le Gia Kiet
- eed5002a Dao Minh Huy

List of present students

- 0a3bf046 Tran Thu Trang
- 1b602ae4 Nguyen Thi Quynh Anh
- 24a0d615 Hong Anh
- 4143e26d Hong Phuc
- 423cd2d4 Tran Thi Lan
- 6e288648 Nguyen Thanh Hoa

Figure 4. 13. Attendance tracking function UI

Room
List of classrooms

Select a room to view schedule

ENG1	40
ENG2	45
ENG3	50
ENG4	40
ENG5	45
ENG6	40
ENG7	45
ENG8	50

Figure 4. 14. Room page

The screenshot shows the English Center Management software interface. On the left, there's a sidebar with icons for General, Courses, Classes, Rooms (which is selected), and Users. Below the sidebar, it says "admin Manager" and has a "Sign out" button. The main area is titled "Room" and "List of classrooms". It features a search bar and a list of rooms with their capacities: ENG1 (40), ENG2 (45), ENG3 (50), ENG4 (40), ENG5 (45), ENG6 (40), ENG7 (45), and ENG8 (50). To the right is a weekly schedule grid from Monday to Saturday. The grid shows sessions: Session 1 (empty), Session 2 (empty), Session 3 (ENG3, ENG3, ENG3), and Session 4 (empty). The "Adv IELTS" course is listed for Session 3.

Figure 4. 15. Viewing schedule for each room function UI

A modal dialog titled "Room Information" with the sub-instruction "Fill in the data for room." It contains fields for Room ID (8214a197), Name (placeholder: Enter the room name), and Capacity (placeholder: Enter capacity). A purple "Submit" button is at the bottom.

Figure 4. 16. Updating room's information function UI

User ID	Account Name	Password	Display Name	Role	Action
b413de1a	admin	*****	admin	Manager	
6dc24a08	jakid	*****	Le Gia Kiet	Student	
0a5d3b96	dangtrenduong	*****	Dao Tien Dat	Student	
d70b9243	huykrito	*****	Dao Minh Huy	Student	
4e61fcf9	mrphan	*****	Phan Chi Cuong	Student	
18a9e586	lequo	*****	Le Quoc Khanh	Student	
4a20e3f3	phamthao	*****	Pham Thao	Student	
e3e9c0a8	quangminh	*****	Quang Minh	Student	

Show Password

Figure 4. 17. Manage Users page

Enroll in a Class

Fill in the data for enrollment.

User ID
21f6bbc9

Account ID Display name
0a5d3b96 Dao Tien Dat

Class
Choose a Class ▾

Submit

Figure 4. 18. Enroll in a class function UI

X

User Information

Fill in the data for user.

User ID

Username

Password

Full name

Role

Submit

Figure 4. 19. User information updating function UI

4.2.3. Description about functions of teacher

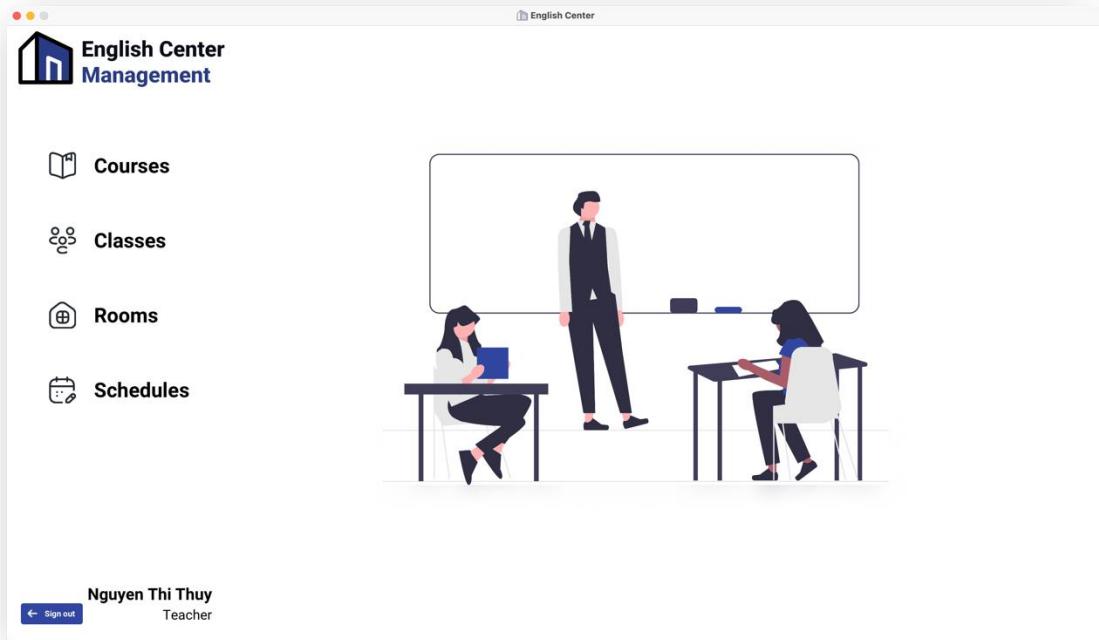


Figure 4. 20. Home page

Figure 4. 21. Courses page

The screenshot shows a software application window titled "English Center Management". The left sidebar contains navigation links: "Courses", "Classes" (which is selected and highlighted in blue), "Rooms", and "Schedules". Below these is a user profile for "Nguyen Thi Thuy" (Teacher). The main content area is titled "Classes" and "List of classes". It features a search bar and a table with the following data:

Class ID	Class Name	Course Name	Teacher	Room	Start date	Finish Date
7fc90be3	Intro TOEIC	Introduction to TOEIC	Nguyen Thi Thuy	ENG1	15-Jun-22	15-Jul-22
3fd22f17	IELTS Writing	IELTS Writing Masterclass	Nguyen Thi Thuy	ENG8	12-Apr-22	12-May-22
17e05a58	Eng Vocab	Basic English Vocabulary Builder	Nguyen Thi Thuy	ENG10	30-Sep-22	30-Oct-22
f3e19b7f	Bus Eng Comm	Business English Communication	Nguyen Thi Thuy	ENG15	11-Dec-22	11-Jan-23

Figure 4. 22. Classes page

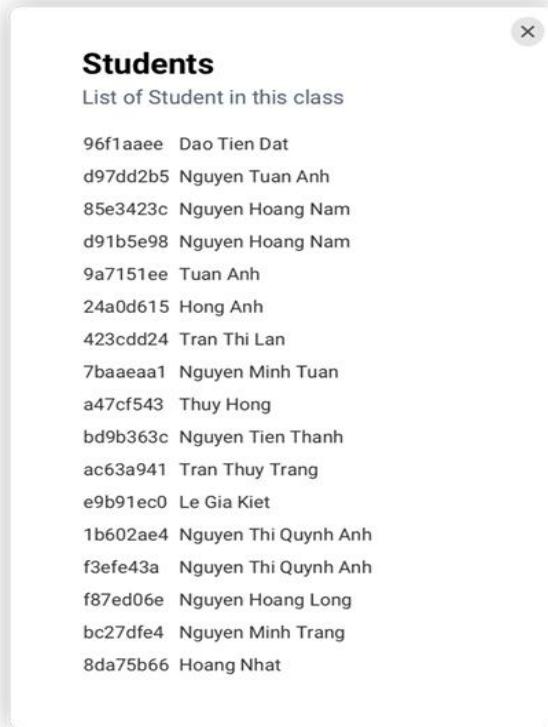


Figure 4. 23. Viewing student list of a class function UI

The screenshot shows the "English Center Management" software interface. On the left, there is a sidebar with icons for "Courses", "Classes" (which is selected), "Rooms", and "Schedules". The main area is titled "Lesson" with the subtitle "Lessons in Class". It displays a list of lessons under the heading "Intro TOEIC" and "Introduction to TOEIC". The lessons are:

- Listening Section Overview** (Listening) - 15-Jun-22
- Part 1: Photographs** (Listening) - 16-Jun-22
- Part 2: Question-Response** (Listening) - 17-Jun-22
- Part 3: Conversations** (Listening) - 18-Jun-22
- Part 4: Short Talks** (Listening) - 19-Jun-22
- Writing Section Overview** (Writing) - 20-Jun-22
- Part 5: Sentence Complet...** (Writing)

On the right, there are tabs for "Materials" and "Attendances". The "Materials" tab shows a list of PDF files with their paths:

- /Users/choingo/Documents/HK4/materials_app/7319a09d/Listening part1.pdf
- /Users/choingo/Documents/HK4/materials_app/7319a09d/Listening final.pdf
- /Users/choingo/Documents/HK4/materials_app/7319a09d/Listening part6.pdf
- /Users/choingo/Documents/HK4/materials_app/7319a09d/Listening part3.pdf
- /Users/choingo/Documents/HK4/materials_app/7319a09d/Listening part5.pdf
- /Users/choingo/Documents/HK4/materials_app/7319a09d/Listening part1.pdf
- /Users/choingo/Documents/HK4/materials_app/7319a09d/Listening part2.pdf

A yellow arrow points to the first item in the list.

Figure 4. 24. Lesson page

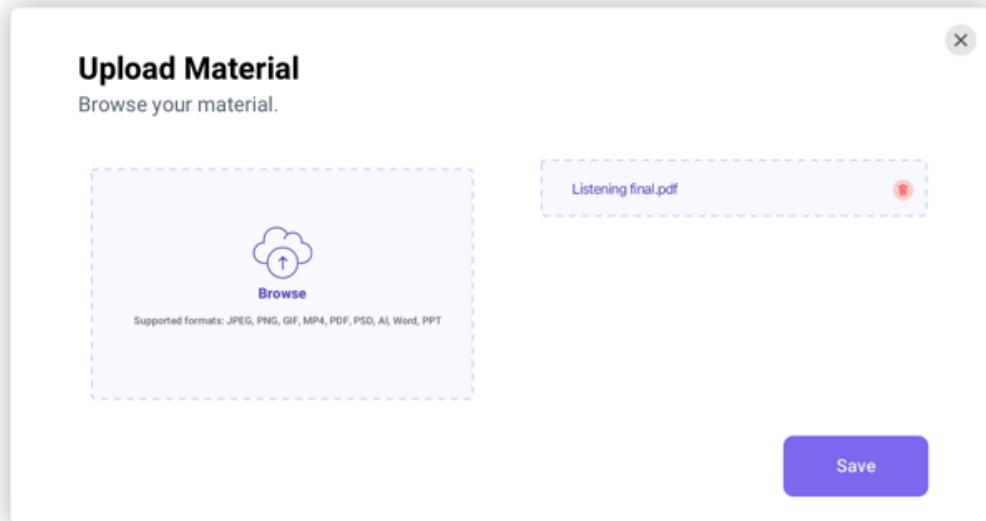


Figure 4. 25. Uploading materials function UI

Lesson
Lessons in Class

Intro TOEIC
Introduction to TOEIC

Section	Type	Date
Listening Section Overview	Listening	15-Jun-22
Part 1: Photographs	Listening	16-Jun-22
Part 2: Question-Response	Listening	17-Jun-22
Part 3: Conversations	Listening	18-Jun-22
Part 4: Short Talks	Listening	19-Jun-22
Writing Section Overview	Writing	20-Jun-22
Part 5: Sentence Complet...	Writing	

Absent 5

Total 25

List of absent students:

- 7baaea1 Nguyen Minh Tuan
- e9b91ec0 Le Gia Kiet
- eed5002a Dao Minh Huy
- f715eab7 Hiep Nhu
- f87ed06e Nguyen Hoang Long

List of present students:

- 0a3bf046 Tran Thu Trang
- 1b602ae4 Nguyen Thi Quynh Anh
- 24a0d615 Hong Anh
- 4143e26d Hong Phu
- 423cd24 Tran Thi Lan
- 6e288648 Nguyen Thanh Hoa
- 85e3423c Nguyen Hoang Nam
- 8da75b66 Hoang Nhat
- 96f1aaee Dao Tien Dat
- 9a7151ee Tuan Anh
- a47cf543 Thuy Hong
- ac65a941 Tran Thuy Trang
- ad4726d0 Thanh Truc
- b7b2075a Tran Thuy Trang
- bc27df64 Nguyen Minh Trang
- bd9b363c Nguyen Tien Thanh
- cd11h5e08 Nguyen Hoang Nam

Figure 4. 26. Attendance tracking function UI

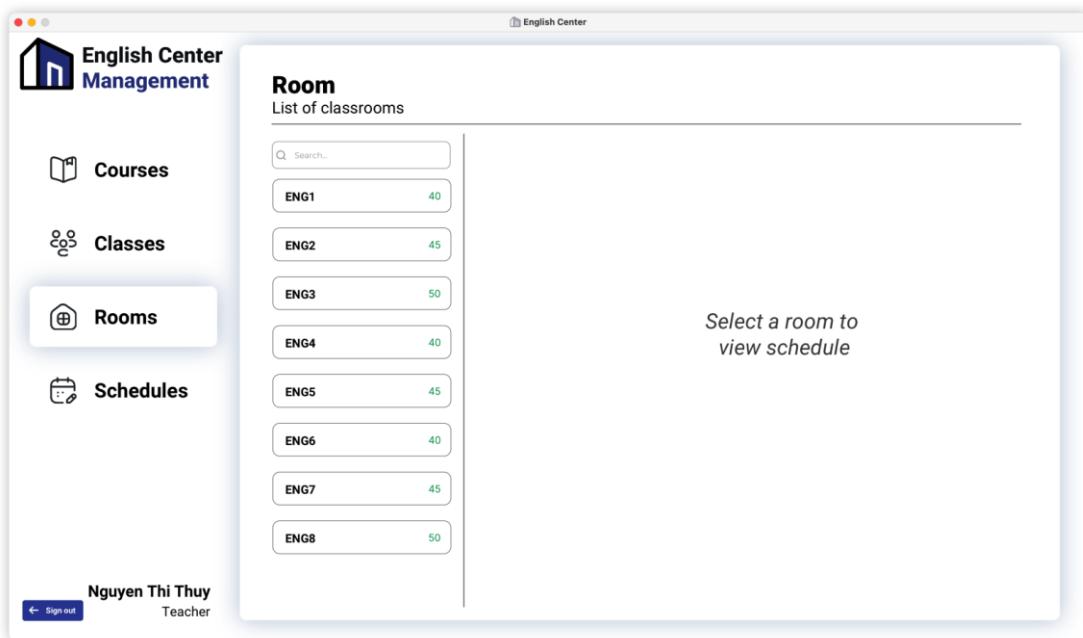


Figure 4. 27. Room page

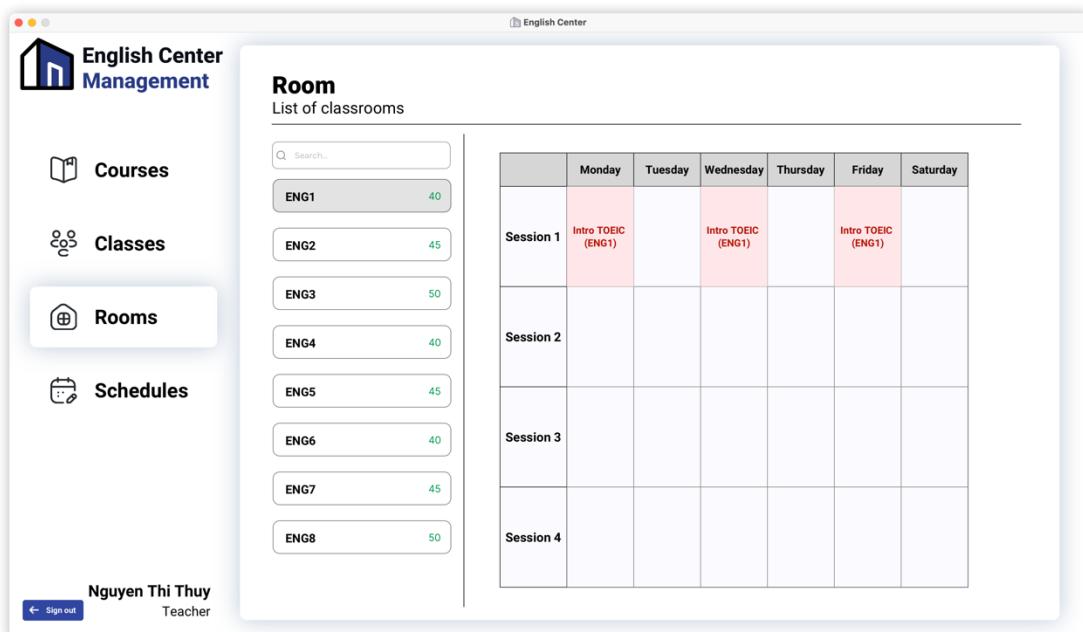


Figure 4. 28. Viewing Schedule of each room function UI

The screenshot shows a software interface titled "English Center Management". On the left sidebar, there are four options: "Courses", "Classes", "Rooms", and "Schedules", with "Schedules" being the active tab. The main area is titled "Schedule" and "Schedule for user". Below this is a weekly grid from Monday to Saturday, divided into four sessions (Session 1 to Session 4). The grid shows various English classes scheduled across the days.

	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
Session 1	Intro TOEIC (ENG1)		Intro TOEIC (ENG1)		Intro TOEIC (ENG1)	
Session 2		Eng Vocab (ENG10)		Eng Vocab (ENG10)		Eng Vocab (ENG10)
Session 3	Bus Eng Comm (ENG15)		Bus Eng Comm (ENG15)		Bus Eng Comm (ENG15)	
Session 4		IELTS Writing (ENG8)		IELTS Writing (ENG8)		IELTS Writing (ENG8)

At the bottom left, it says "Nguyen Thi Thuy Teacher". There is also a "Sign out" button.

Figure 4. 29. Schedule page

4.2.4. Description about functions of student

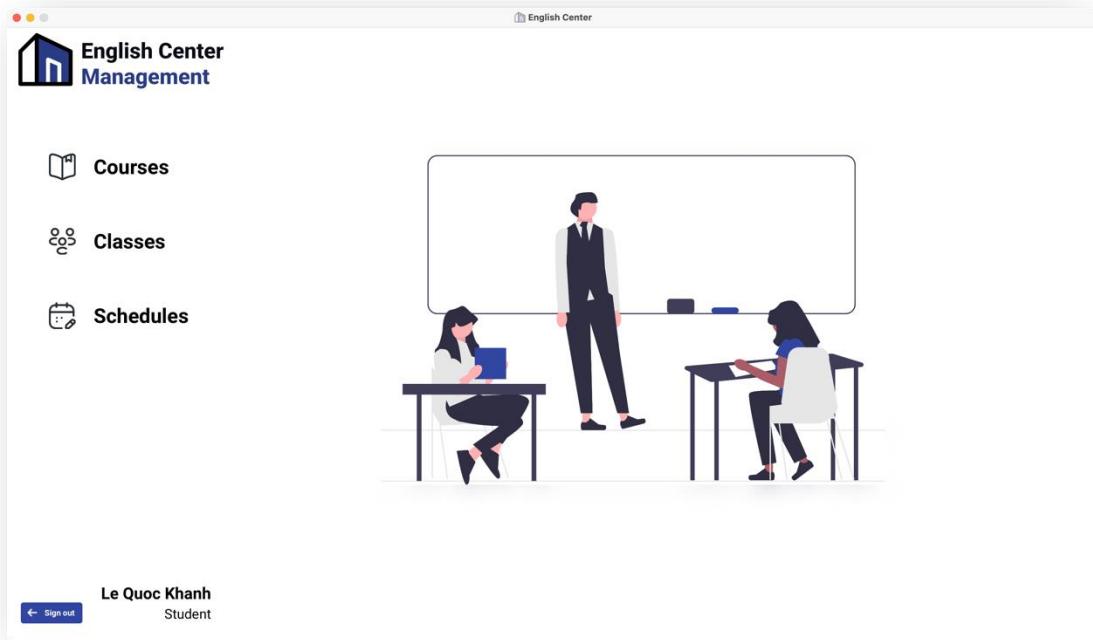


Figure 4. 30. Home page

Course Title	Chapters	Level
Introduce to International English	21 Chapters	Intermediate
English Pronunciation	28 Chapters	Intermediate
English Grammar Essential	28 Chapters	Intermediate
Basic English Vocabulary	28 Chapters	Beginner
Introduction to TOEIC Test	19 Chapters	Beginner
Conversational English	28 Chapters	Beginner
Business English Communication	21 Chapters	Advanced
IELTS Writing Master	28 Chapters	Intermediate

Figure 4. 31. Courses page

The screenshot shows the English Center Management software interface. On the left, there is a sidebar with icons for Courses, Classes, and Schedules. The user is signed in as Le Quoc Khanh, a Student. The main content area is titled "Chapters" and "Manage Chapters in a Course". It displays a course titled "Introduction to TOEIC" with a purple header. The course has 19 chapters, all labeled "Beginner". A search bar is at the top right. Below the course title, three sections are listed: "Part 2: Describe a Picture", "Part 5: Sentence Completion", and "Part 6: Error Recognition". Each section includes a brief description and a category: "Speaking" for the first, "Writing" for the second, and "Writing" for the third.

Figure 4. 32. Chapters page

The screenshot shows the English Center Management software interface. On the left, there is a sidebar with icons for Courses, Classes, and Schedules. The user is signed in as Le Quoc Khanh, a Student. The main content area is titled "Classes" and "List of classes". It features a search bar at the top. Below it is a table listing five classes:

Class ID	Class Name	Course Name	Teacher	Room	Start date	Finish Date
3fd22f17	IELTS Writing	IELTS Writing Masterclass	Nguyen Thi Thuy	ENG8	12-Apr-22	12-May-22
17e05a58	Eng Vocab	Basic English Vocabulary Builder	Nguyen Thi Thuy	ENG10	30-Sep-22	30-Oct-22
d3e82192	Intro TOEIC	Introduction to TOEIC	Tieu Quang Minh	ENG11	08-Jan-22	08-Feb-22
a2f8db10	Eng Grammar	English Grammar Essentials	Le Anh Tuan	ENG4	05-Mar-22	05-Apr-22

Figure 4. 33. Classes page

The screenshot shows the English Center Management system interface. On the left, there's a sidebar with icons for Courses, Classes, and Schedules. The Classes icon is highlighted. Below it, the user information 'Le Quoc Khanh Student' and a 'Sign out' button are visible. The main content area is titled 'Lesson' and shows a list of 'Lessons in Class' under the heading 'IELTS Writing' and 'IELTS Writing Masterclass'. The lessons are listed with their titles, categories (e.g., 'Writing'), and dates (e.g., 12-Apr-22 to 17-Apr-22). To the right, there's a 'Materials' section with a search bar and two PDF files listed: 'Writing task2.pdf' and 'Writing task1.pdf'. A yellow arrow points from the text 'Materials' towards the PDF files.

Figure 4. 34. Lesson page

The screenshot shows the English Center Management system interface. The sidebar on the left has 'Courses' and 'Classes' icons, with 'Classes' being the active tab. Below it is the user information 'Le Quoc Khanh Student' and a 'Sign out' button. The main content area is titled 'Schedule' and shows a 'Schedule for user' grid. The grid has rows labeled 'Session 1' through 'Session 4' and columns labeled Monday through Saturday. The schedule details are as follows:

	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
Session 1						
Session 2		Eng Vocab (ENG10)		Eng Vocab (ENG10)		Eng Vocab (ENG10)
Session 3	Intro TOEIC (ENG11)		Intro TOEIC (ENG11)		Intro TOEIC (ENG11)	
Session 4		Eng Grammar (ENG4)		Eng Grammar (ENG4)		Eng Grammar (ENG4)

Figure 4. 35. Schedule page

Chapter 5: CONCLUSION

5.1. Project achievement

The team has successfully developed an English center management system with a user-friendly interface, fast responsiveness, and a diverse range of useful functions that meet customer requirements. The following are the main functionalities of our system:

- Login/Sign Out
- Manage Courses
- Manage Classes
- Manage Classrooms
- Manage Users
- Export class report
- Analyze
- Explore Courses
- View Schedule/Classes/Classrooms/Lessons/Materials

By utilizing Java, JavaFX, and Oracle technologies, we constructed the system. The implementation of the MVC (Model-View-Controller) design pattern played a crucial role in dividing our project into multiple modules, leading to seamless problem-solving in project management.

5.2. Limitations

In the challenging journey of project management, the team has experienced valuable and profound moments. Each project is not just a task, but also a heartfelt and hopeful journey.

Together, the team has built a united collective, with each member contributing significantly to bring the project to fruition. The feeling of pride spreads throughout the space as the efforts of each individual are recognized and honored.

However, there are also times when the work is not highly evaluated or even faced with failure, creating a mixed emotion within the team. But through those failures, the team never loses hope. Instead, the team sees those failures as valuable lessons to become stronger and more optimistic for future projects.

The lessons learned from experiences to conclude some limitations in the project:

- Data recovery and backup function hasn't been completed.
- Management of enrollment and placement test hasn't been completed.
- Didn't pay too much attention to teacher & student information management.
- Academic results management hasn't been completed.
- Revenue, expenditure and finance management hasn't been completed.
- The interface is not compatible with diverse devices.

Due to limited deployment time and difficulties in exploring new technologies, lack of extensive documentation, some mistakes cannot be avoided. We greatly appreciate any feedback and guidance from you to make the project more complete.

5.3. Development direction

Firstly, our team will solve the limitations above, including Data recovery and backup function and Making the app more native with other devices. In order to continuously improve and meet the demands of computerized management of short-term training courses, in the future, we will strive to study more thoroughly, delve deeper, and better complete the project.

- System optimization: Continuously enhance the existing management system by optimizing the source code, improving performance and security, ensuring stable and reliable system operation, while enhancing user experience.

- Feature and module expansion: Consider developing and integrating new features and modules into the system. For example, it may include a schedule management feature, online interaction between teachers and students, or integration of artificial intelligence technology to improve the learning process.
- Towards a multi-platform system: Develop mobile applications or responsive interfaces to support users in accessing and utilizing the English center on various platforms. This provides flexibility and convenience for students to access content and interact with the system.
- Research and application of new technologies: Monitor and research emerging technology trends in the field of education and apply them to the management system of the English center. Consider the use of artificial intelligence, machine learning, or other new technologies to enhance the learning process and center management.