



RadiantIQ



UNIVERSITÀ  
DI TRENTO

Department of  
Information Engineering and Computer Science

**Project's acronym:** RADIANTIQ

**Project's title:** RadiantIQ

**Start date:** 08/03/2024

**Finish date:** DD/MM/YYYY

## D1 - RadiantIQ Software Specification

**WP1:** Software Specification

**Task 1.1:** Define Software Specification

**Submission date:** 09/04/2024

**Responsible:** RadiantIQ

**Version:** 1.0

**Status:** Completed

**Author(s):** Lorenzo Cattai, Gabriele Pernici, Anh Tu Duong, Lorenzo Negut

**Deliverable type:** DOCUMENT



## Version list:

Version	Authors	Date	Description
0.1	Anh Tu Duong	09/04/2024	Define the scheme of the document
1.0	Anh Tu Duong	11/04/2024	Complete domain analysis and objectives

## Contents

<b>1</b>	<b>Introduction</b>	<b>6</b>
<b>2</b>	<b>Domain Analysis</b>	<b>7</b>
2.1	Domain: Interactive Education	7
2.2	SWOT analysis	7
<b>3</b>	<b>Project Objectives</b>	<b>8</b>
3.1	Use of minigames in interactive learning	8
3.2	Use AI to help develop a more compelling learning experience	8
3.3	Make learning science more approachable and enjoyable	8
3.4	Provide single-topic focused content	8
3.5	Provide private classes as instances of a course	8
3.6	Allow better student-professor interaction	8
3.7	Provide quality guarantees on the material published	8
3.8	Collect data and provide a progress history	8
<b>4</b>	<b>Actors</b>	<b>9</b>
4.1	Roles functioning	9
<b>5</b>	<b>Functional Requirements</b>	<b>10</b>
<b>6</b>	<b>Non-functional Requirements</b>	<b>11</b>
<b>7</b>	<b>Use Cases</b>	<b>12</b>
7.1	Tables	12
7.1.1	Login	12
7.1.2	Logout	13
7.1.3	Credential recovery	13
7.1.4	Registration	14
7.1.5	Delete Account	14
7.1.6	Modify account's core settings	15
7.1.7	Modify account's secondary settings	16
7.1.8	Modify AI theming	16
7.1.9	Access profile and statistics	17
7.1.10	Change user role	17
7.1.11	Create course	18
7.1.12	Modify course	18
7.1.13	Delete course	19
7.1.14	Archive course	19
7.1.15	View course	19
7.1.16	Review course	20
7.1.17	Create class	20
7.1.18	Modify class	21
7.1.19	Terminate class	21
7.1.20	Archive class	22
7.1.21	View class	22
7.1.22	Display class' attendees	22
7.1.23	Display class' statistics	23
7.1.24	Join class	23
7.1.25	Leave class	23
7.1.26	Publish article	24
7.1.27	Modify article	24
7.1.28	Delete article	25
7.1.29	Archive article	25
7.1.30	View article	25



7.1.31	Review article . . . . .	26
7.1.32	Create minigame . . . . .	26
7.1.33	Modify minigame . . . . .	27
7.1.34	Delete minigame . . . . .	27
7.1.35	Archive minigame . . . . .	28
7.1.36	Register minigame from developer . . . . .	28
7.1.37	Register minigame from observer . . . . .	29
7.1.38	Pay developer . . . . .	29
7.1.39	Use tech support chat . . . . .	30
7.1.40	Use development chat . . . . .	30
7.1.41	Search element . . . . .	31
7.1.42	Remove review . . . . .	31
7.2	Diagrams . . . . .	32
7.2.1	Account and general purpose . . . . .	33
7.2.2	Article . . . . .	34
7.2.3	Chat . . . . .	35
7.2.4	Class . . . . .	36
7.2.5	Course . . . . .	37
7.2.6	Minigame . . . . .	38
<b>8</b>	<b>Context Diagram . . . . .</b>	<b>39</b>
<b>9</b>	<b>Components Diagram . . . . .</b>	<b>40</b>
<b>10</b>	<b>Class Diagram . . . . .</b>	<b>41</b>
<b>11</b>	<b>Conclusion . . . . .</b>	<b>42</b>



## Acronyms

Acronym	Description
FRS	Functional Requirement Specification
FSD	Functional Specification Document
LLMs	Large Language Models



## 1 Introduction

The project consists in a platform providing a better learning experience for scientific subjects. The idea is to include standard formal explanations of topics (with associated exercises) accompanied by a small number of interactive minigames. To better involve the students, each exercise will be put in an AI generated context (e.i. a physics problem related to speeds and distances could be told using the story of Achilles and the turtoise). Moreover, the formal explanations can be genrated by an AI, uploaded by a professor or by a combination of the two. Lastly, AI is used to suggest which topics should be revised for the students using the platform.



## 2 Domain Analysis

### 2.1 Domain: Interactive Education

We want to create an application in the education domain and try to make it extremely interactive.

### 2.2 SWOT analysis

- **Strengths:**

- (1) Use of AI to generate basic information in the articles, with the supervision of admins.
- (1) Use of AI to generate more advanced information from user's customized necessities and weaknesses
- (2) Use a more fun and interactive approach in the education process
- (2) Learn by trying with minigames with immediate feedback or in a more standard way with articles
- (3) Possibility for experts to contribute with their own articles
- FLOSS Education Platform

- **Weaknesses:**

- Developing effective mini-games' experiences might be time consuming from a developer perspective
- Making fun and yet instructive experiences is hard
- Careful management of the AI generated information is needed
- Quality check on article uploaded must be implemented
- Costs of using LLMs

- **Opportunities:**

- (1) Using AI in education is an emerging idea that has not yet spread widely and can make this system unique
- (2) Various studies show the positive effect of interaction and hands-on experiences in the learning process
- Large demand of easy and complete ways to learn science
- (3) Lack of effective and proficient communication/cooperation/interaction between students and teachers which creates an ample improvement margin to be enhanced

- **Threats:**

- Possible limitations of AI technologies from government entities
- Minigames must be entertaining to be successful
- Brilliant is a direct competitor (tho it does not currently use AI nor personalize its exercises and lectures)
- Hard to find funding

## 3 Project Objectives

### 3.1 Use of minigames in interactive learning

Create some minigames to make learning more interactive and more intriguing for students.

### 3.2 Use AI to help develop a more compelling learning experience

Use AI to generate compelling descriptions of the topics, to make learning funnier.

### 3.3 Make learning science more approachable and enjoyable

Make the platform an accessible starting point in the learning of science, to allow everyone to learn science using intuition and reason.

### 3.4 Provide single-topic focused content

Providing single topic courses means the possibility to create a learning path specific for the interests of the user.

### 3.5 Provide private classes as instances of a course

Create classes, from the general courses, to allow teachers/professors to integrate the platform in their standard lectures.

### 3.6 Allow better student-professor interaction

Using the classes the professors can understand which topics are clearer and then provide feedback. Moreover students can easily determine the level of comprehension of each topic before a test.

### 3.7 Provide quality guarantees on the material published

Have personnel checking the validity of the published material, while not interfering with the private resources.

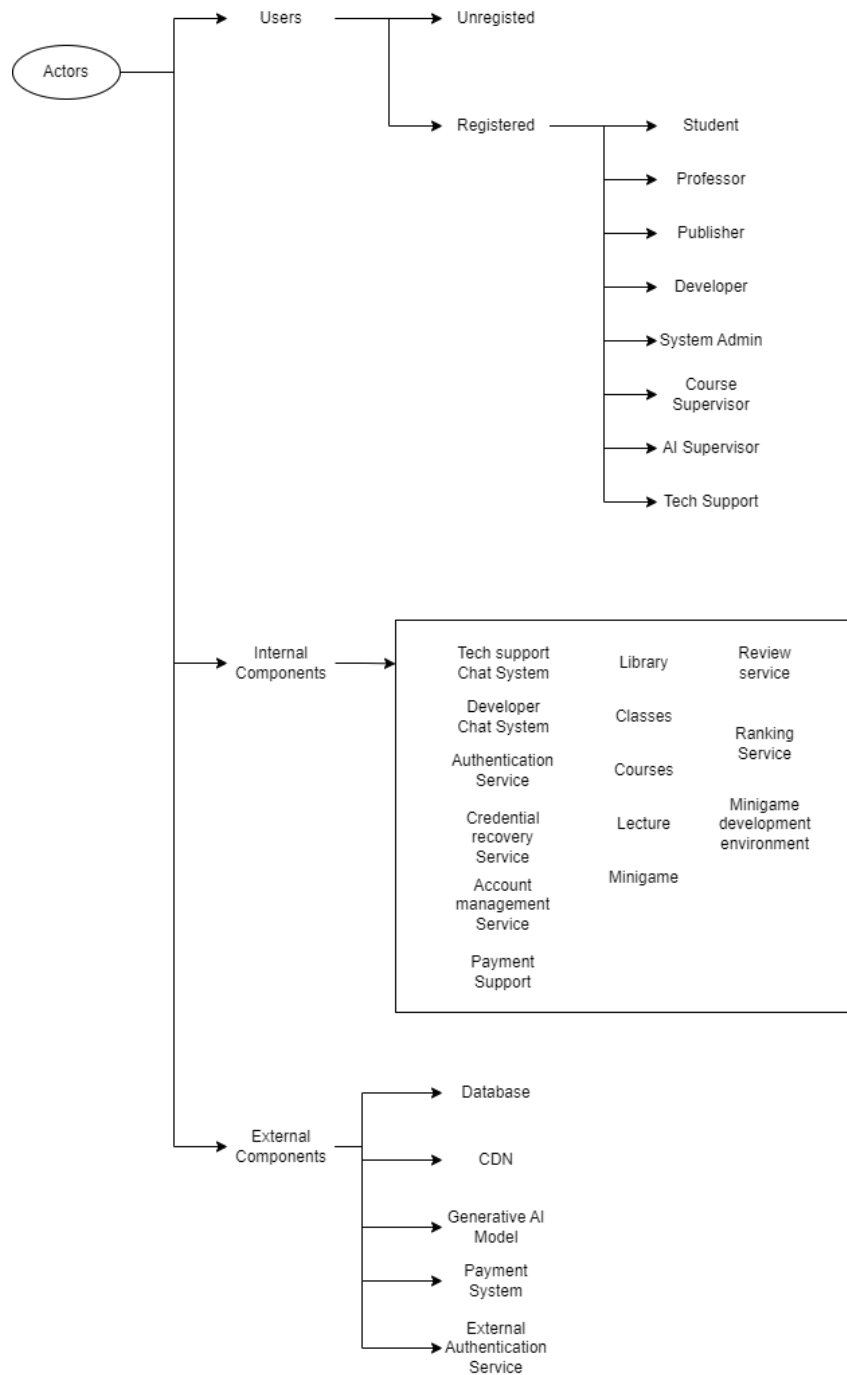
### 3.8 Collect data and provide a progress history

For the learning user, having feedback on the level of comprehension is fundamental and will help them focus more on the less understood topics.



## 4 Actors

We created a Mind map summarizing the actors scheme of the system, it is shown below.



### 4.1 Roles functioning

Each role provides specific and unique functionalities (that are mutually exclusive). Each user can have multiple roles assigned to them simultaneously, in a composite role system similar to the one used in Discord. We chose this model to easily separate functionalities and assign them to users modularly.



## 5 Functional Requirements



## 6 Non-functional Requirements

## 7 Use Cases

We created both the tables for each use case and the complete diagram. Moreover, we created some partial diagrams containing some use cases and organized logically.

### 7.1 Tables

The following are the 42 use case tables.

#### 7.1.1 Login

Login	
<b>ID</b>	UC1 - UC_LOGIN
<b>Actors</b>	Registered users (Students, Professors, Publishers, Developers, AI supervisors, Tech Supports, Course Supervisors, System Admins)
<b>Preconditions</b>	The user is registered and has credentials
<b>Sequence</b>	<ol style="list-style-type: none"> <li>1. The actor selects the login option</li> <li>2. The actor inserts its credentials <ol style="list-style-type: none"> <li>2.1 If the actor uses an external authentication system it's redirected</li> </ol> </li> <li>3. Credentials are verified</li> </ol>
<b>Postconditions</b>	The user is authenticated and can access its roles' privileges
<b>Alternative sequence 1</b>	<ol style="list-style-type: none"> <li>1. The actor inputs the wrong password for the first, second or third time</li> </ol>
<b>Postconditions</b>	The user is not authenticated and a notification is sent to the user registered with the inserted username
<b>Alternative sequence 2</b>	<ol style="list-style-type: none"> <li>1. The actor inputs the wrong password for the fifth time</li> </ol>
<b>Postconditions</b>	The user is not allowed to login for a significant time and a notification is sent to the user registered with the inserted username
<b>Alternative sequence 3</b>	<ol style="list-style-type: none"> <li>1. The actor inputs the wrong username</li> </ol>
<b>Postconditions</b>	The user is not authenticated



### 7.1.2 Logout

Logout	
<b>ID</b>	UC2 - UC_LOGOUT
<b>Actors</b>	Registered users (Students, Professors, Publishers, Developers, AI supervisors, Tech Supports, Course Supervisors, System Admins)
<b>Preconditions</b>	The user is registered and logged in
<b>Sequence</b>	<ol style="list-style-type: none"><li>1. The actor selects the logout option</li><li>2. The actor confirms their choice</li></ol>
<b>Postconditions</b>	The user is logged out

### 7.1.3 Credential recovery

Credential recovery	
<b>ID</b>	UC3 - UC_CREDENTIAL_REC
<b>Actors</b>	Registered users (Students, Professors, Publishers, Developers, AI supervisors, Tech Supports, Course Supervisors, System Admins)
<b>Preconditions</b>	The user is registered and has forgotten credentials
<b>Sequence</b>	<ol style="list-style-type: none"><li>1. The actor has forgotten their credentials</li><li>2. The actor requires new credentials</li></ol>
<b>Postconditions</b>	The user acquires new credentials on the previously specified recovery channel

#### 7.1.4 Registration

Registration	
<b>ID</b>	UC4 - UC_REGISTRATION
<b>Actors</b>	Unregistered users
<b>Preconditions</b>	The actor decides to register
<b>Sequence</b>	<ol style="list-style-type: none"> <li>1. The actor selects the registration option</li> <li>2. The actor decides their credentials, core settings and which roles they want to apply for <ol style="list-style-type: none"> <li>2.1 The credentials' compliance with security policies is asserted</li> </ol> </li> </ol>
<b>Postconditions</b>	The user is registered and has now one or more roles assigned
<b>Alternative sequence 1</b>	<ol style="list-style-type: none"> <li>1. The actor wants to apply for the Course Supervisor role</li> <li>2. The actor provides the apposite proof of identity</li> </ol>
<b>Postconditions</b>	The user is registered as a Course Supervisor
<b>Alternative sequence 2</b>	<ol style="list-style-type: none"> <li>1. The actor inputs incomplete, incorrect or unacceptable credentials/proof</li> </ol>
<b>Postconditions</b>	The user is not registered

#### 7.1.5 Delete Account

Delete Account	
<b>ID</b>	UC5 - UC_ACC_DEL
<b>Actors</b>	Registered users (Students, Professors, Publishers, Developers, AI supervisors, Tech Supports, Course Supervisors, System Admins)
<b>Preconditions</b>	The user is registered, logged in and decides to delete the account
<b>Sequence</b>	<ol style="list-style-type: none"> <li>1. The actor selects the unregistration option</li> <li>2. The actor confirms their choice</li> <li>3. The actor confirms their identity by inserting the account's password</li> </ol>
<b>Postconditions</b>	The user is unregistered



### 7.1.6 Modify account's core settings

Modify account's core settings	
<b>ID</b>	UC6 - UC_MOD_CORE_SETT
<b>Actors</b>	Registered users (Students, Professors, Publishers, Developers, AI supervisors, Tech Supports, Course Supervisors, System Admins)
<b>Preconditions</b>	The user is registered, logged in and decides to change one or more of the core settings (e.g. password, username, recovery channel, personal information, ...)
<b>Sequence</b>	<ol style="list-style-type: none"><li>1. The actor selects the modify core settings option</li><li>2. The actor confirms their identity by inserting the account's password</li><li>3. The actor changes the selected settings</li><li>4. The actor confirms the choice</li></ol>
<b>Postconditions</b>	The change in settings is saved
<b>Alternative sequence 1</b>	<ol style="list-style-type: none"><li>1. The actor doesn't confirm their changes or cancels the modification</li></ol>
<b>Postconditions</b>	The settings stay the same

### 7.1.7 Modify account's secondary settings

Modify account's secondary settings	
<b>ID</b>	UC7 - UC_MOD_SEC_SETT
<b>Actors</b>	Registered users (Students, Professors, Publishers, Developers, AI supervisors, Tech Supports, Course Supervisors, System Admins)
<b>Preconditions</b>	The user is registered, logged in and decides to change one or more of the secondary settings (e.g. theme, layout, ...)
<b>Sequence</b>	<ol style="list-style-type: none"> <li>1. The actor selects the modify secondary settings option</li> <li>2. The actor changes the selected settings</li> <li>3. The actor confirms the choice</li> </ol>
<b>Postconditions</b>	The change in settings is saved
<b>Alternative sequence 1</b>	<ol style="list-style-type: none"> <li>1. The actor doesn't confirm their changes or cancels the modification</li> </ol>
<b>Postconditions</b>	The settings stay the same

### 7.1.8 Modify AI theming

Modify AI theming	
<b>ID</b>	UC8 - UC_MOD_AI_THEMING
<b>Actors</b>	Students, AI supervisors, Tech Supports
<b>Preconditions</b>	The user is registered and logged in
<b>Sequence</b>	<ol style="list-style-type: none"> <li>1. The user selects the AI theming option</li> <li>2. The user inputs a prompt for the AI theming</li> <li>3. The user confirms the modification</li> </ol>
<b>Postconditions</b>	The theming prompt is modified
<b>Alternative sequence 1</b>	<ol style="list-style-type: none"> <li>1. The user doesn't confirm the modification</li> </ol>
<b>Postconditions</b>	AI theming continues with the previous prompt





### 7.1.9 Access profile and statistics

Access profile and statistics	
<b>ID</b>	UC9 - UC_ACCESS_PROFILE
<b>Actors</b>	Registered users (Students, Professors, Publishers, Developers, AI supervisors, Tech Supports, Course Supervisors, System Admins)
<b>Preconditions</b>	The user is registered and logged in
<b>Sequence</b>	1. The actor selects the account display option
<b>Postconditions</b>	The profile with all its statistics is displayed

### 7.1.10 Change user role

Change user role	
<b>ID</b>	UC10 - UC_USER_ROLE
<b>Actors</b>	Registered users (Students, Professors, Publishers, Developers, AI supervisors, Tech Supports, Course Supervisors, System Admins)
<b>Preconditions</b>	The user is registered and logged in
<b>Sequence</b>	1. The actor selects "User Role" panel 2. The actor selects the one of the user role from the list
<b>Postconditions</b>	The user role changed following by his dashboard role

### 7.1.11 Create course

Create course	
<b>ID</b>	UC11 - UC_COURSE_CREATE
<b>Actors</b>	Publishers
<b>Preconditions</b>	The user is registered and logged in
<b>Sequence</b>	<ol style="list-style-type: none"> <li>1. The actor selects the course panel</li> <li>2. The actor selects the “Create new course” option</li> <li>3. The actor fills all the mandatory sections for creating a new course</li> <li>4. The actor confirms to create new course</li> </ol>
<b>Postconditions</b>	The course is created successfully
<b>Alternative sequence 1</b>	<ol style="list-style-type: none"> <li>1. The actor doesn't fill all the mandatory sections for creating a new course</li> <li>2. The actor confirms to save the unfinished work</li> </ol>
<b>Postconditions</b>	The course is saved as draft

### 7.1.12 Modify course

Modify course	
<b>ID</b>	UC12 - UC_COURSE_MOD
<b>Actors</b>	Professors, Publishers, AI supervisors, Course Supervisors, System Admins
<b>Preconditions</b>	The user is registered and logged in
<b>Sequence</b>	<ol style="list-style-type: none"> <li>1. The actor selects the course</li> <li>2. The actor selects the “Modify course” option</li> <li>3. The actor modifies the course</li> <li>4. The actor confirms the modification</li> </ol>
<b>Postconditions</b>	The course is updated successfully
<b>Alternative sequence 1</b>	<ol style="list-style-type: none"> <li>1. The actor doesn't confirm the modification</li> </ol>
<b>Postconditions</b>	The course keeps its previous state



### 7.1.13 Delete course

Delete course	
<b>ID</b>	UC13 - UC_DEL_COURSE
<b>Actors</b>	Publishers, Course Supervisors, System Admins
<b>Preconditions</b>	The user is registered and logged in
<b>Sequence</b>	<ol style="list-style-type: none"><li>1. The actor selects “Delete course”</li><li>2. The actor confirms the deletion</li></ol>
<b>Postconditions</b>	The course is deleted

### 7.1.14 Archive course

Archive course	
<b>ID</b>	UC14 - UC_ARC_COURSE
<b>Actors</b>	Publishers, Course Supervisors, System Admins
<b>Preconditions</b>	The user is registered and logged in
<b>Sequence</b>	<ol style="list-style-type: none"><li>1. The actor selects “Archive course”</li><li>2. The actor confirms the modification</li></ol>
<b>Postconditions</b>	The course is moved to archive

### 7.1.15 View course

View course	
<b>ID</b>	UC15 - UC_COURSE_VIEW
<b>Actors</b>	Registered users (Students, Professors, Publishers, Developers, AI supervisors, Tech Supports, Course Supervisors, System Admins) and Unregistered users
<b>Preconditions</b>	The user is registered, logged in and has the right to enter the course
<b>Sequence</b>	<ol style="list-style-type: none"><li>1. The actor select a course from the dashboard or from a library</li></ol>
<b>Postconditions</b>	The course is displayed, along with the global ranking if any minigame is present



### 7.1.16 Review course

Review course	
<b>ID</b>	UC16 - UC_COURSE_REV
<b>Actors</b>	Registered users (Students, Professors, Publishers, Developers, AI supervisors, Tech Supports, Course Supervisors, System Admins)
<b>Preconditions</b>	The user is registered, logged in and has the right to enter the course. Moreover the course is public and opened
<b>Sequence</b>	<ol style="list-style-type: none"><li>1. The actor leave a review (comment) for the course</li></ol>
<b>Postconditions</b>	The review is added to the course

### 7.1.17 Create class

Create class	
<b>ID</b>	UC17 - UC_CLASS_CREATE
<b>Actors</b>	Professors
<b>Preconditions</b>	The user is registered and logged in
<b>Sequence</b>	<ol style="list-style-type: none"><li>1. The actor selects the class panel</li><li>2. The actor selects the "Create new class" option</li><li>3. The actor fills all the mandatory sections for creating a new class</li><li>4. The actor has possibility to invite Student(s) to the class</li><li>5. The actor confirms to create new class</li></ol>
<b>Postconditions</b>	The class is created successfully



### 7.1.18 Modify class

Modify class	
<b>ID</b>	UC18 - UC_CLASS_MOD
<b>Actors</b>	Professors, System Admins
<b>Preconditions</b>	The user is registered, logged in and has sufficient permissions (owns the class or is admin)
<b>Sequence</b>	<ol style="list-style-type: none"><li>1. The actor selects the class</li><li>2. The actor selects the “Modify class” option</li><li>3. The actor modifies the class</li><li>4. The actor has possibility to invite Student(s) to the class</li><li>5. The actor confirms the modification</li></ol>
<b>Postconditions</b>	The class is updated successfully
<b>Alternative sequence 1</b>	<ol style="list-style-type: none"><li>1. The actor doesn't confirm the modification</li></ol>
<b>Postconditions</b>	The class remains its previous state

### 7.1.19 Terminate class

Terminate class	
<b>ID</b>	UC19 - UC_CLASS_TERM
<b>Actors</b>	Professors, System Admins
<b>Preconditions</b>	The user is registered, logged in and has sufficient permissions (owns the class or is admin)
<b>Sequence</b>	<ol style="list-style-type: none"><li>1. The actor selects “Terminate class”</li><li>2. The actor confirms the termination</li></ol>
<b>Postconditions</b>	The class is terminated, but all the information about the class remains public



### 7.1.20 Archive class

Archive class	
<b>ID</b>	UC20 - UC_ARC_CLASS
<b>Actors</b>	Professors, System Admins
<b>Preconditions</b>	The user is registered, logged in and has sufficient permissions (owns the class or is admin)
<b>Sequence</b>	<ol style="list-style-type: none"><li>1. The actor selects "Archive class"</li><li>2. The actor confirms the modification</li></ol>
<b>Postconditions</b>	The class is moved to archive and all the information about the class is accessible only for Professors and System admins

### 7.1.21 View class

View class	
<b>ID</b>	UC21 - UC_CLASS_VIEW
<b>Actors</b>	Registered users (Students, Professors, Publishers, Developers, AI supervisors, Tech Supports, Course Supervisors, System Admins)
<b>Preconditions</b>	The user is registered and logged in
<b>Sequence</b>	<ol style="list-style-type: none"><li>1. The actor selects a class from the dashboard or from an invitation</li></ol>
<b>Postconditions</b>	The public information of the class is displayed

### 7.1.22 Display class' attendees

Display class' attendees	
<b>ID</b>	UC22 - UC_CLASS_ATTENDEES
<b>Actors</b>	Students, Professors
<b>Preconditions</b>	The user is registered, logged in and enrolled in a class
<b>Sequence</b>	<ol style="list-style-type: none"><li>1. The actor selects a class they are enrolled into</li></ol>
<b>Postconditions</b>	The list of people attending the class is displayed

### 7.1.23 Display class' statistics

Display class' statistics	
<b>ID</b>	UC23 - UC_CLASS_PERFORMANCE
<b>Actors</b>	Professors
<b>Preconditions</b>	The user is registered, logged in and manages a class
<b>Sequence</b>	1. The actor selects a class they manage
<b>Postconditions</b>	The performance and statistics of all the attendees is displayed

### 7.1.24 Join class

Join class	
<b>ID</b>	UC24 - UC_CLASS_JOIN
<b>Actors</b>	Students
<b>Preconditions</b>	The user is registered, logged in and has the right to join the class. Moreover the class is opened and public
<b>Sequence</b>	1. The actor selects "Join class" option
<b>Postconditions</b>	The actor now joined the class. The actor's information, statistics and progresses for the class is initialized and is public for the class's owner

### 7.1.25 Leave class

Leave class	
<b>ID</b>	UC25 - UC_CLASS_LEAVE
<b>Actors</b>	Students
<b>Preconditions</b>	The user is registered, logged in and is part of a class
<b>Sequence</b>	1. The actor selects "Leave class" option 2. The actor confirm their choice
<b>Postconditions</b>	The actor leaves the class, but their information, statistics and progresses for the class is saved and is still public for the class's owner



### 7.1.26 Publish article

Publish article	
<b>ID</b>	UC26 - UC_ART_PUB
<b>Actors</b>	Publisher
<b>Preconditions</b>	The user is registered and logged in
<b>Sequence</b>	<ol style="list-style-type: none"><li>1. The actor selects the article panel</li><li>2. The actor selects the “Publish new article” option</li><li>3. The actor fills all the mandatory sections for publishing a new article</li><li>4. The actor confirms to publish new article</li></ol>
<b>Postconditions</b>	The article is published successfully

### 7.1.27 Modify article

Modify article	
<b>ID</b>	UC27 - UC_ART_MOD
<b>Actors</b>	Publisher, Course Supervisors, System Admins
<b>Preconditions</b>	The user is registered and logged in
<b>Sequence</b>	<ol style="list-style-type: none"><li>1. The actor selects the article</li><li>2. The actor selects the “Modify article” option</li><li>3. The actor modifies the article or its visibility</li><li>4. The actor confirms the modification</li></ol>
<b>Postconditions</b>	The article is updated successfully
<b>Alternative sequence 1</b>	<ol style="list-style-type: none"><li>1. The actor doesn't confirm the modification</li></ol>
<b>Postconditions</b>	The article remains in its previous state





### 7.1.28 Delete article

Delete article	
<b>ID</b>	UC28 - UC_ART_DEL
<b>Actors</b>	Publisher, Course Supervisors, System Admins
<b>Preconditions</b>	The user is registered, logged in and has ownership of the article (or is admin)
<b>Sequence</b>	<ol style="list-style-type: none"> <li>1. The actor selects "Delete article"</li> <li>2. The actor confirms the deletion</li> </ol>
<b>Postconditions</b>	The article is deleted

### 7.1.29 Archive article

Archive article	
<b>ID</b>	UC29 - UC_ART_ARC
<b>Actors</b>	Publisher, Course Supervisors, System Admins
<b>Preconditions</b>	The user is registered, logged in and has ownership of the article (or is admin)
<b>Sequence</b>	<ol style="list-style-type: none"> <li>1. The actor selects "Archive article"</li> <li>2. The actor confirms the modification</li> </ol>
<b>Postconditions</b>	The article is moved to archive

### 7.1.30 View article

View article	
<b>ID</b>	UC30 - UC_ART_VIEW
<b>Actors</b>	Registered users (Students, Professors, Publishers, Developers, AI supervisors, Tech Supports, Course Supervisors, System Admins) and Unregistered users
<b>Preconditions</b>	The user is registered and logged in
<b>Sequence</b>	<ol style="list-style-type: none"> <li>1. The actor selects an article from the dashboard</li> </ol>
<b>Postconditions</b>	The article is opened

### 7.1.31 Review article

Review article	
<b>ID</b>	UC31 - UC_ART_REV
<b>Actors</b>	Registered users (Students, Professors, Publishers, Developers, AI supervisors, Tech Supports, Course Supervisors, System Admins)
<b>Preconditions</b>	The user is registered and logged in. The article is opened and public
<b>Sequence</b>	<ol style="list-style-type: none"> <li>1. The actor leave a review (comment) for the article</li> </ol>
<b>Postconditions</b>	The review is added to the article

### 7.1.32 Create minigame

Create minigame	
<b>ID</b>	UC32 - UC_MINIGAME_CREATE
<b>Actors</b>	Developers
<b>Preconditions</b>	The user is registered and logged in
<b>Sequence</b>	<ol style="list-style-type: none"> <li>1. The actor selects the developer panel</li> <li>2. The actor selects the "Create new minigame" option</li> <li>3. The actor uses the environment to create the minigame</li> <li>4. The actor confirms to create new minigame</li> </ol>
<b>Postconditions</b>	The minigame is created successfully and saved to minigame storage

### 7.1.33 Modify minigame

Modify minigame	
<b>ID</b>	UC33 - UC_MINIGAME_MOD
<b>Actors</b>	Developers, Course Supervisors, System Admin, Tech Supports
<b>Preconditions</b>	The user is registered and logged in
<b>Sequence</b>	<ol style="list-style-type: none"> <li>1. The actor selects the minigame</li> <li>2. The actor selects the “Modify minigame” option</li> <li>3. The actor modifies the minigame</li> <li>4. The actor confirms the modification</li> </ol>
<b>Postconditions</b>	The minigame is updated successfully
<b>Alternative sequence 1</b>	<ol style="list-style-type: none"> <li>1. The actor doesn’t confirm the modification</li> </ol>
<b>Postconditions</b>	The minigame remains in its previous state

### 7.1.34 Delete minigame

Delete minigame	
<b>ID</b>	UC34 - UC_MINIGAME_DEL
<b>Actors</b>	Developers, Course Supervisors, System Admin
<b>Preconditions</b>	The user is registered, logged in and owns the minigame (or is admin)
<b>Sequence</b>	<ol style="list-style-type: none"> <li>1. The actor selects “Delete minigame”</li> <li>2. The actor confirms the deletion</li> </ol>
<b>Postconditions</b>	The minigame is deleted from memory and from all the courses containing it



### 7.1.35 Archive minigame

Archive minigame	
<b>ID</b>	UC35 - UC_MINIGAME_ARC
<b>Actors</b>	Developers, Course Supervisors, System Admin
<b>Preconditions</b>	The user is registered, logged in and owns the minigame (or is admin)
<b>Sequence</b>	<ol style="list-style-type: none"><li>1. The actor selects “Archive minigame”</li><li>2. The actor confirms the modification</li></ol>
<b>Postconditions</b>	The minigame is moved to archive and is deleted from all the courses containing it

### 7.1.36 Register minigame from developer

Register minigame from developer	
<b>ID</b>	UC36 - UC_MINIGAME_REG_DEV
<b>Actors</b>	Developers
<b>Preconditions</b>	The user is registered and logged in
<b>Sequence</b>	<ol style="list-style-type: none"><li>1. The actor selects the minigame</li><li>2. The actor selects “Register minigame to course”</li><li>3. The actor chooses the course(s) to register (add) minigame into</li><li>4. The actor fills all the mandatory sections for registering a new minigame to course</li><li>5. The actor confirms to register minigame</li></ol>
<b>Postconditions</b>	The minigame is register successfully to the course(s)



### 7.1.37 Register minigame from observer

Register minigame from observer	
<b>ID</b>	UC37 - UC_MINIGAME_REG
<b>Actors</b>	Publishers, Developers, AI supervisors, Tech Supports, Course Supervisors, System Admins
<b>Preconditions</b>	The user is registered, logged in and has authorization from minigame owner
<b>Sequence</b>	<ol style="list-style-type: none"><li>1. The actor selects the course</li><li>2. The actor selects the “Modify course” option</li><li>3. The actor selects “Register minigame to course”</li><li>4. The actor chooses the minigame to add to course</li><li>5. The actor fills all the mandatory sections for registering a new minigame to course</li><li>6. The actor confirms to register minigame</li></ol>
<b>Postconditions</b>	The minigame is register successfully to the course(s)

### 7.1.38 Pay developer

Pay developer	
<b>ID</b>	UC38 - UC_PAY
<b>Actors</b>	Publishers
<b>Preconditions</b>	The user is registered, logged in and has tasked a developer with a minigame
<b>Sequence</b>	<ol style="list-style-type: none"><li>1. The actor selects the payment option</li><li>2. The user selects the external payment method they want to use</li><li>3. The actor follows the external payment system iter</li></ol>
<b>Postconditions</b>	The actor has paid the developer



### 7.1.39 Use tech support chat

Use tech support chat	
<b>ID</b>	UC39 - UC_SUPPORT_CHAT
<b>Actors</b>	Registered users (Students, Professors, Publishers, Developers, AI supervisors, Tech Supports, Course Supervisors, System Admins)
<b>Preconditions</b>	The user is registered, logged in and has a technical problem
<b>Sequence</b>	<ol style="list-style-type: none"><li>1. The actor opens the tech support chat</li><li>2. The user sends/receives a message on the chat</li></ol>
<b>Postconditions</b>	The actor interacts with Tech Supports and starts solving the technical problem

### 7.1.40 Use development chat

Use development chat	
<b>ID</b>	UC40 - UC_DEV_CHAT
<b>Actors</b>	Publishers, Developers
<b>Preconditions</b>	The user is registered, logged in and has the need to discuss about minigame creation
<b>Sequence</b>	<ol style="list-style-type: none"><li>1. The actor opens the development chat</li><li>2. The actor communicates with the commissioner/developer</li></ol>
<b>Postconditions</b>	The actor interacts with Developer/commissioner and starts the minigame development process



#### 7.1.41 Search element

Search element	
<b>ID</b>	UC41 - UC_SEARCH
<b>Actors</b>	Registered users (Students, Professors, Publishers, Developers, AI supervisors, Tech Supports, Course Supervisors, System Admins) and Unregistered users
<b>Preconditions</b>	The user wants to find some material on the platform (course, class, mini-game, lecture, ...)
<b>Sequence</b>	<ol style="list-style-type: none"><li>1. The actor selects the search option</li><li>2. The actor inputs the possible search parameters (name, subject, type, ...)</li></ol>
<b>Postconditions</b>	A list of resources adhering to the parameters is displayed

#### 7.1.42 Remove review

Remove review	
<b>ID</b>	UC42 - UC_DEL_REV
<b>Actors</b>	System Admins
<b>Preconditions</b>	The user is registered and logged in. The chosen review violates some policies of the platform
<b>Sequence</b>	<ol style="list-style-type: none"><li>1. The actor removes review from any material</li></ol>
<b>Postconditions</b>	The review is removed entirely

## 7.2 Diagrams

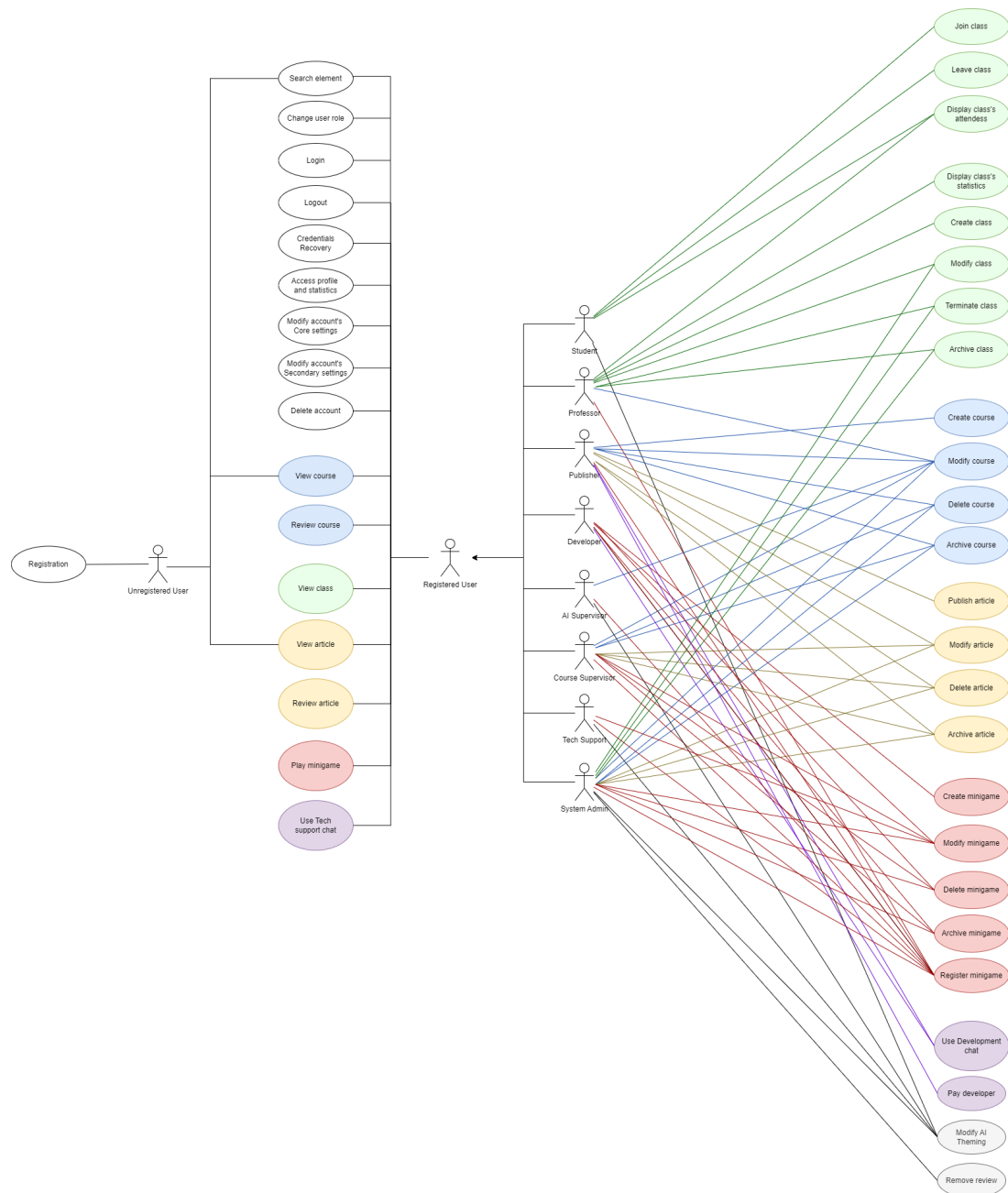


Figure 2: Complete use case diagram



### 7.2.1 Account and general purpose

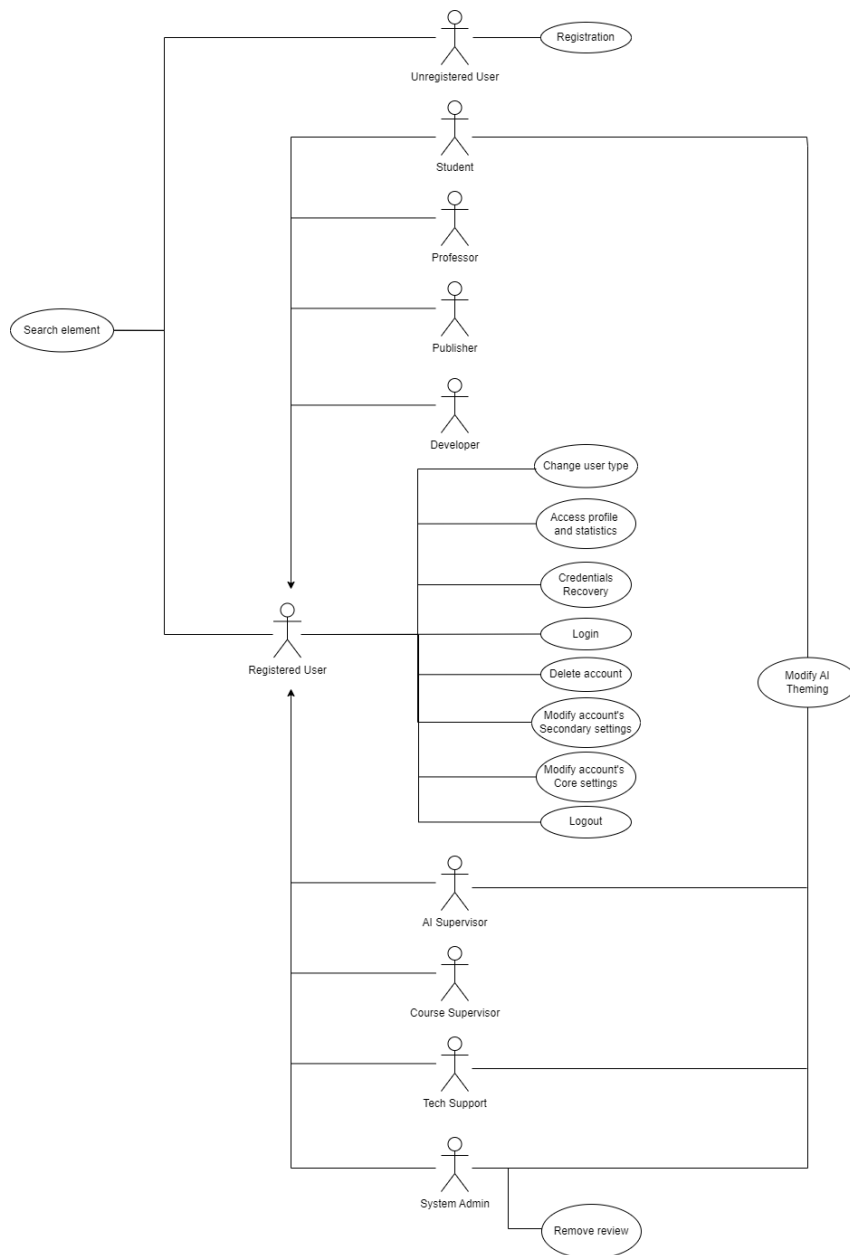


Figure 3: Use case diagram for account system

## 7.2.2 Article

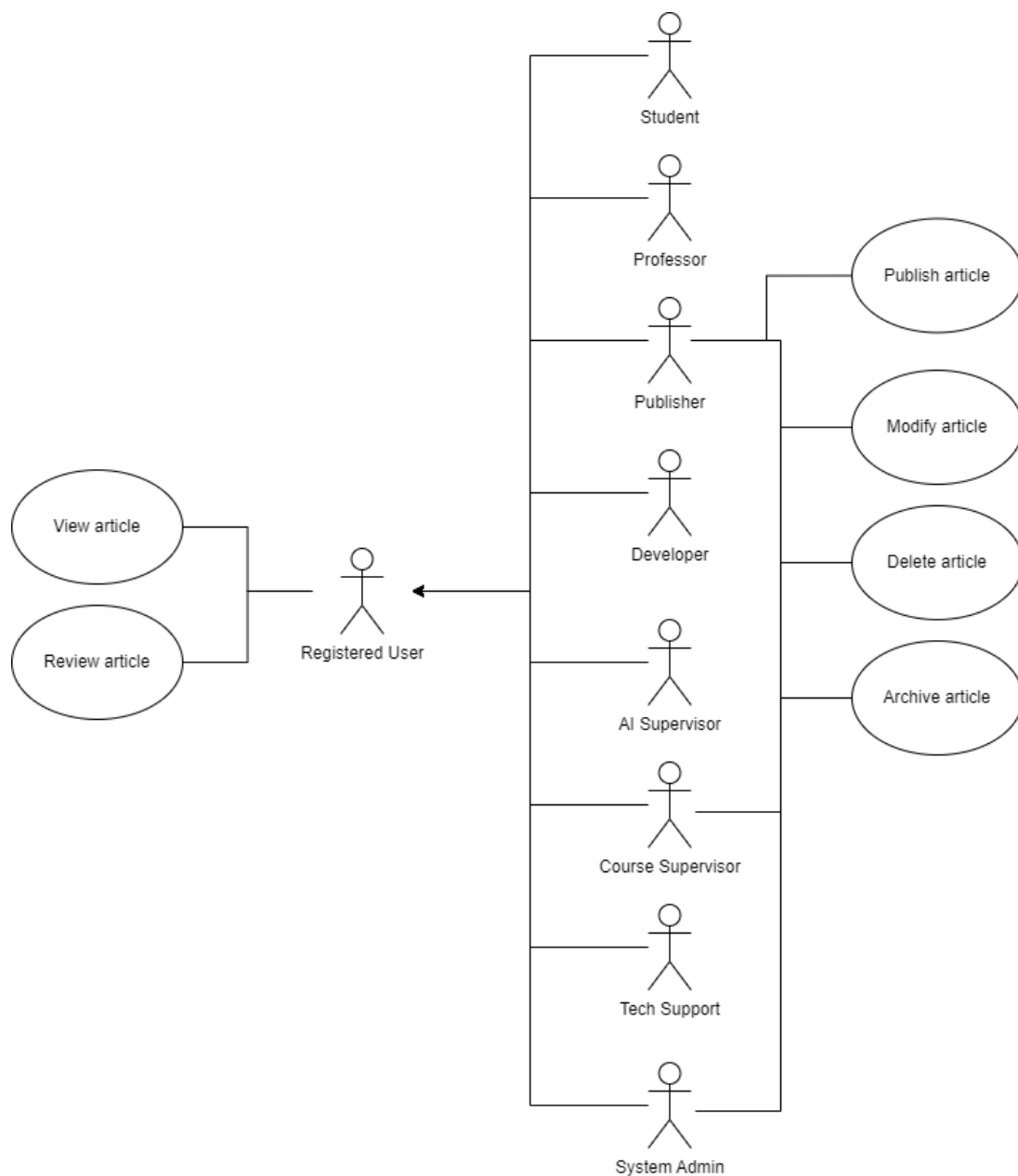


Figure 4: Use case diagram for article system

### 7.2.3 Chat

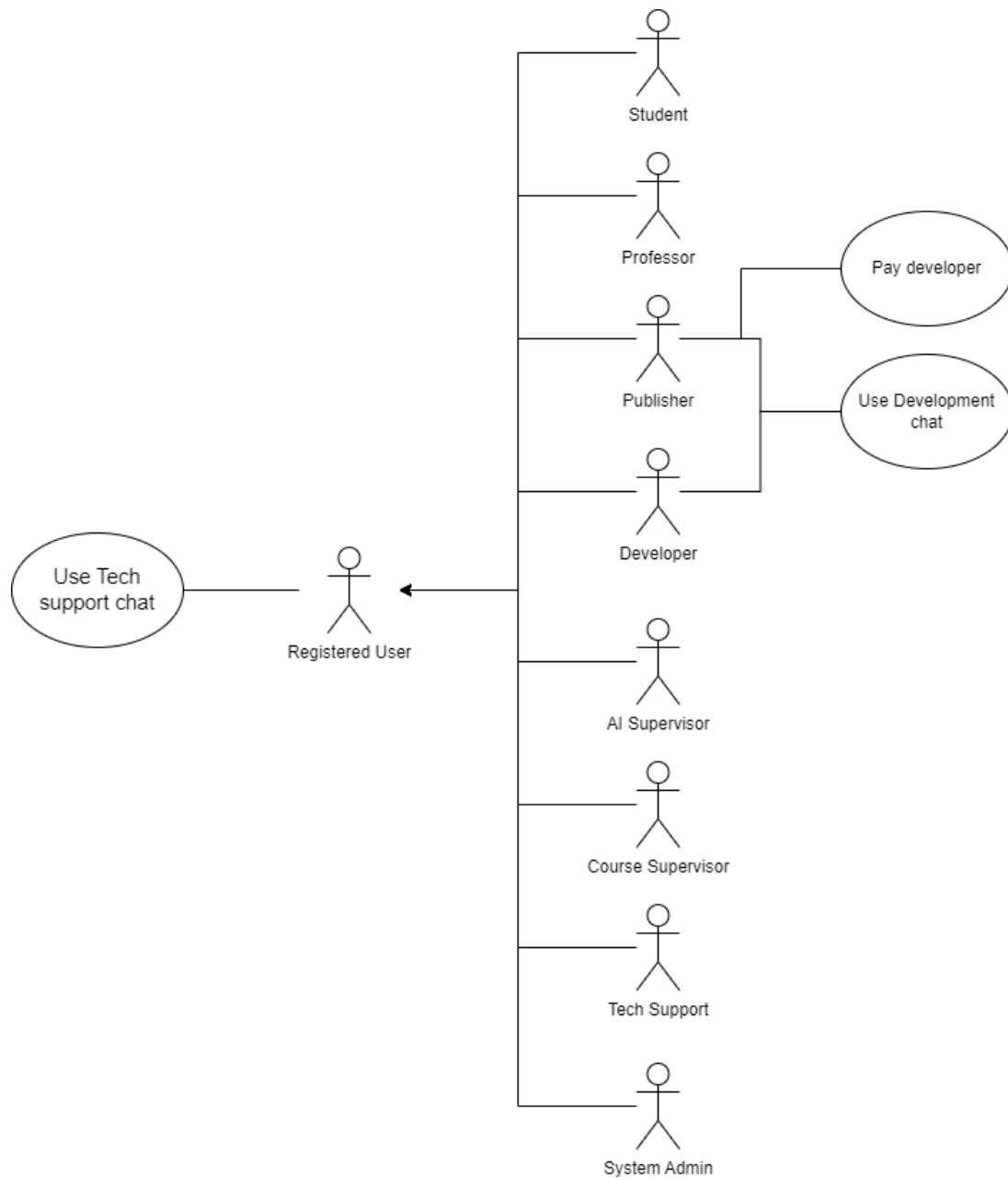


Figure 5: Use case diagram for chat system

## 7.2.4 Class

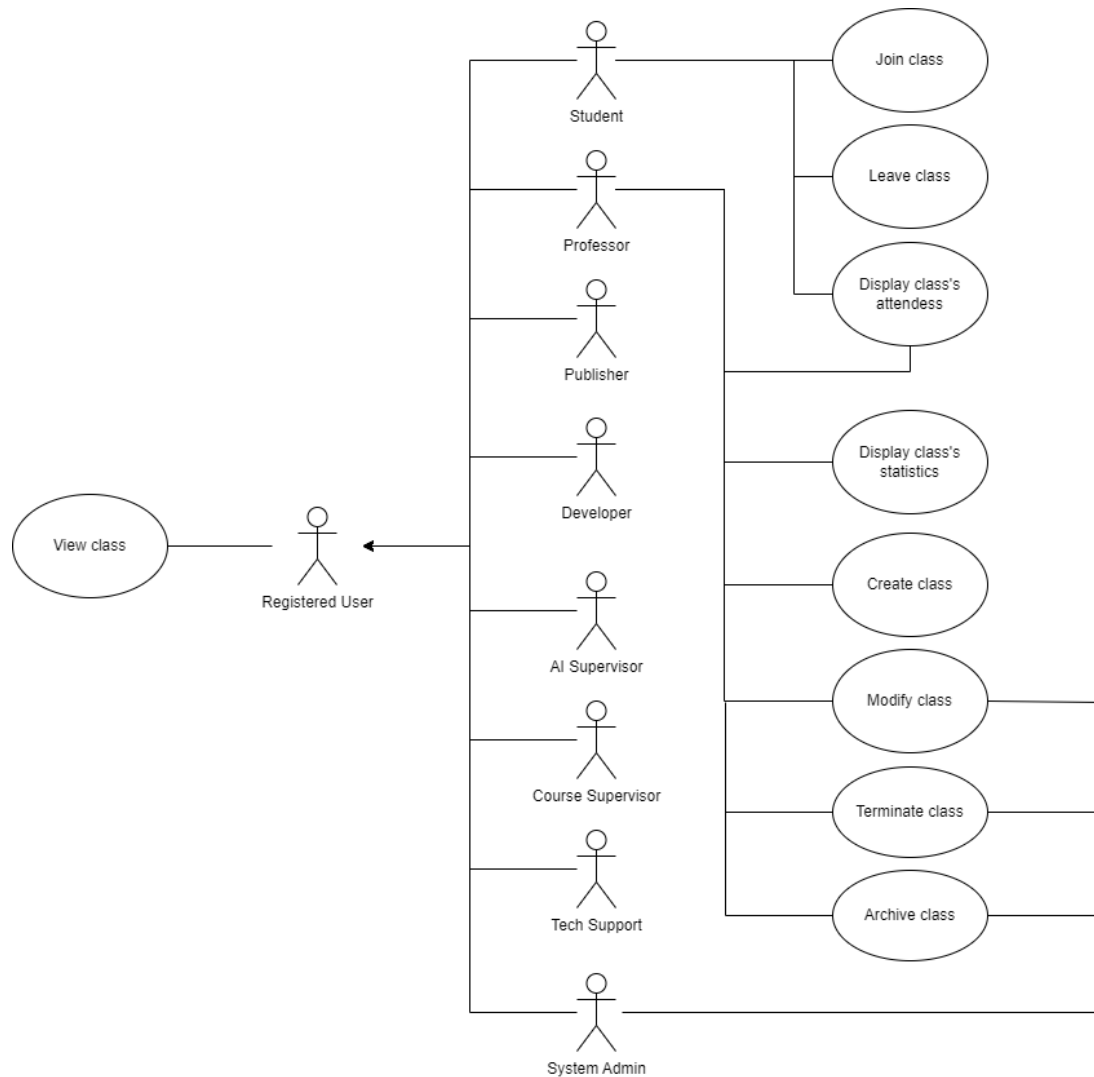


Figure 6: Use case diagram for class system

## 7.2.5 Course

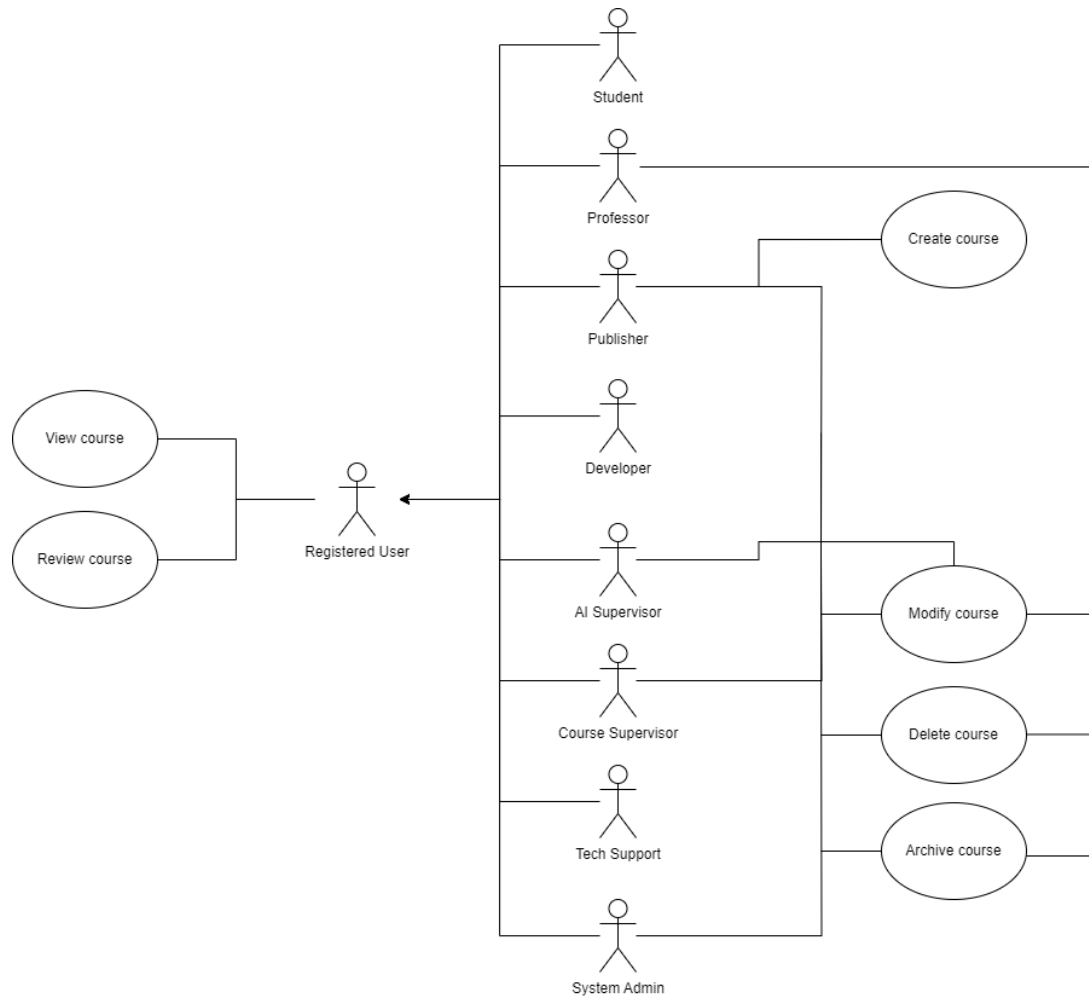
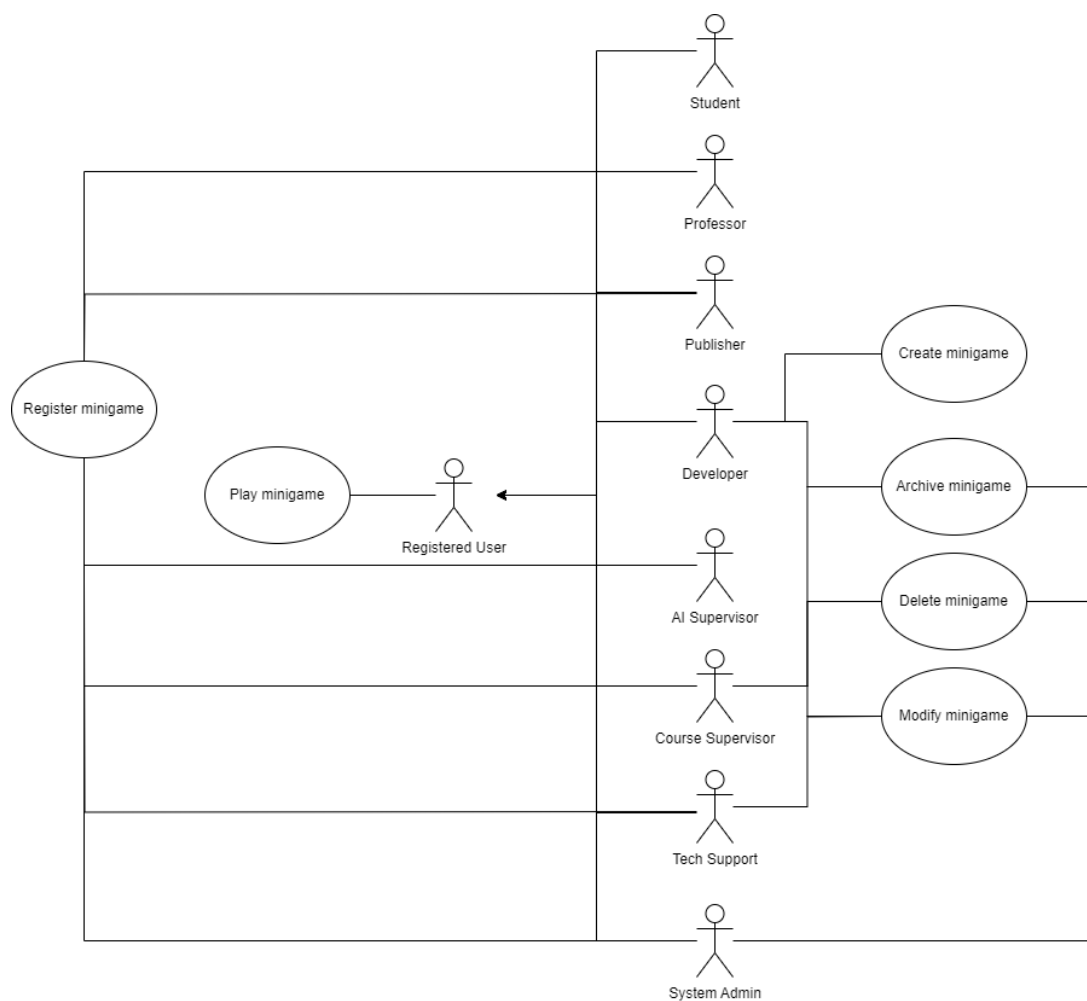
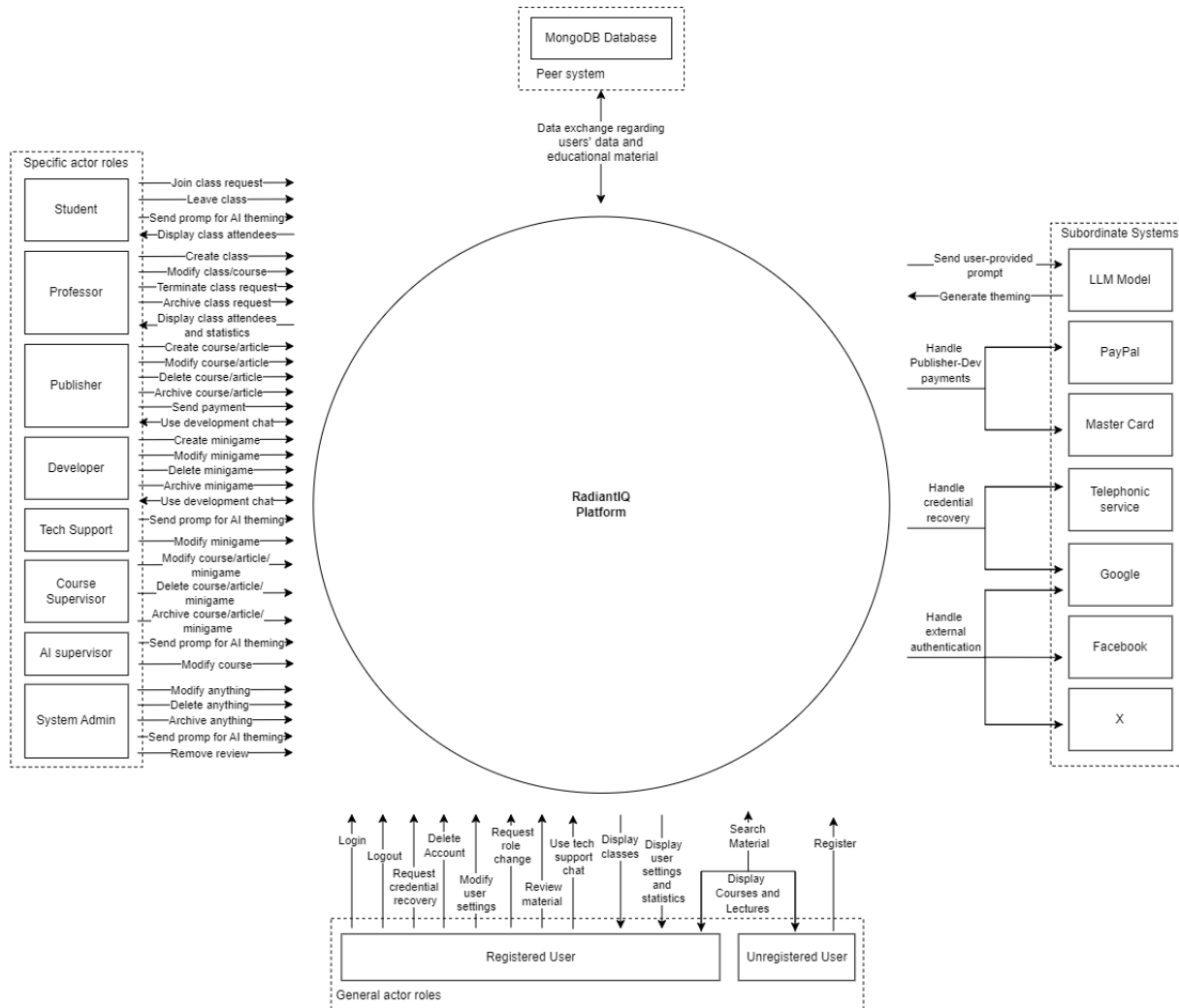


Figure 7: Use case diagram for course system



## 8 Context Diagram

We created a context diagram separating the general roles (registered and unregistered users), the specific roles (all the others), the subsystems (payment, authentication, credential recovery, LLM) and the peer system (database).



## 9 Components Diagram

A components diagram was create for better visualization about all the main components of the system.

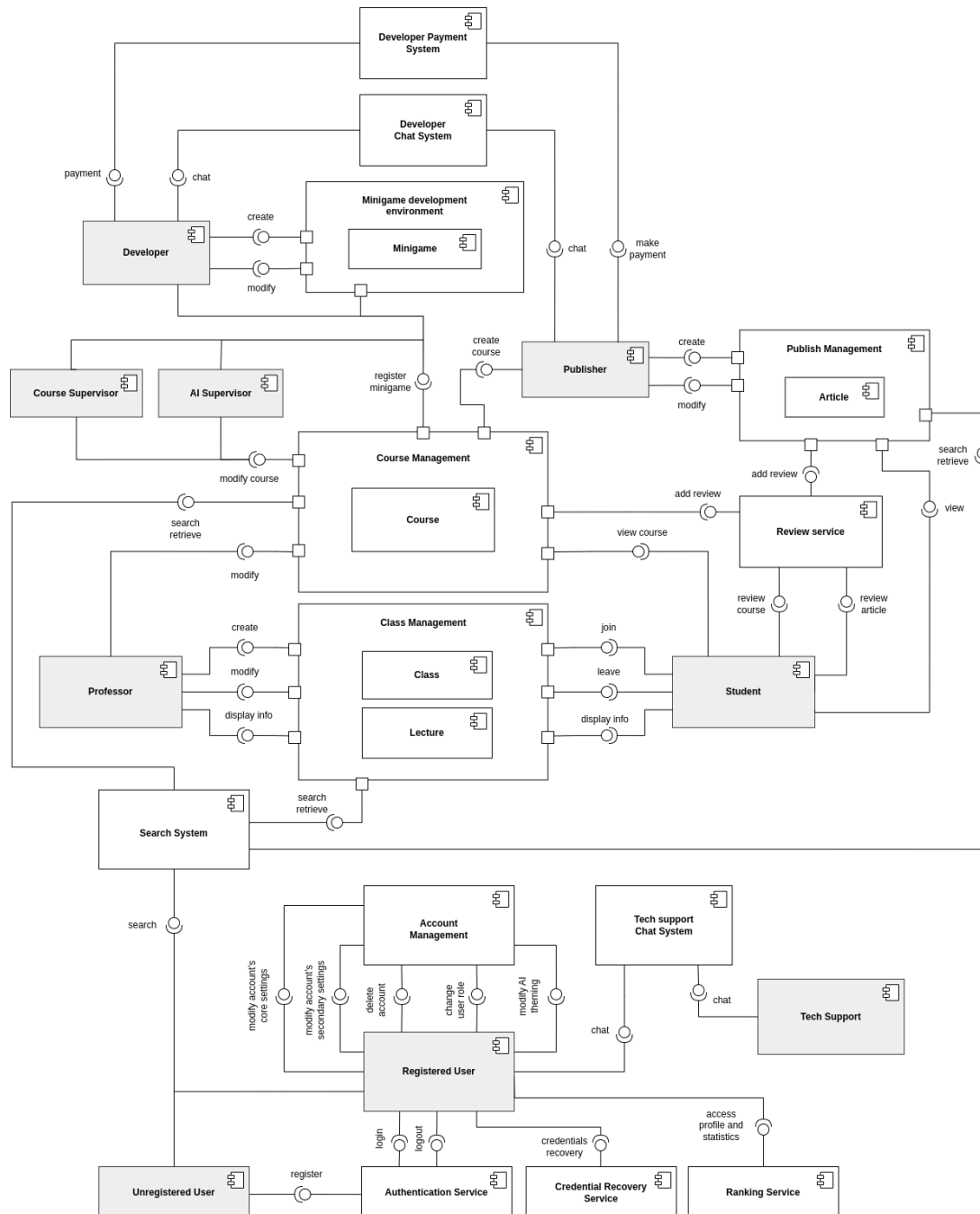


Figure 9: RadiantIQ - Components diagram





## 10 Class Diagram



## 11 Conclusion