Software Development Plan

April 4, 2024

Version 1.1

Presented To:

Dr. Mohamed Ramadan

Submitted By:

Logina Mohamed Fekry Abdelmobdy

REVISION HISTORY

Date	Author	Distributed to	Version
03/10/24	All	All	1.0
04/4/24	All	All	1.1

1. PRODUCT DESCRIPTION

Introducing our online education platform, where students and teachers come together to engage in a seamless learning experience. Our platform is designed with the aim of facilitating learning and teaching in a dynamic virtual environment.

At the heart of our platform lies the empowerment of students. Students have the autonomy to choose courses tailored to their interests and needs. Whether it's a specialized course with a specific teacher or a trending topic in today's academic landscape, our platform offers a diverse range of options. Additionally, students can save their chosen courses for easy access, utilize the search feature to discover new topics, and engage with course content through comments and likes.

For teachers, our platform provides a robust toolkit to enhance their teaching capabilities. Teachers can create and curate playlists filled with educational content and videos, fostering an immersive learning experience for students. Furthermore, teachers can actively monitor and respond to student feedback, fostering a collaborative and supportive learning environment.

Both students and teachers enjoy the flexibility to modify, update, and delete content within their respective roles, ensuring a tailored and dynamic educational experience for all users.

Join us on our journey to revolutionize online education, where learning knows no boundaries and knowledge is just a click away.

2. TEAM DESCRIPTION

Team Members	Logina Mohamed
Concepts	
HTML	X
CSS	X
Java Script	X
PHP	
My SQL	X
Security (hash tables)	

Unit Testing	
Research	X

The skills needed for this project are:

- Time Management
- Good Searching Skills
- Experience in programming
- Experience in database management
- Always willing to learn

In general, I have prior experience in front-end languages: HTML, CSS, and JavaScript. I completed a project on these languages during my studies in information technology.

As for databases, I have a strong foundation, which has been further reinforced by extensive training at ITI.

However, I lack experience in the backend part, particularly in PHP, the language that supports it. I aim to strengthen this skill through continued learning and development.

Unit testing is covered in the software session, and I am currently learning about it.

On the other hand, I have limited experience with APIs and hash tables. I plan to start researching and learning more about them soon.

3. SOFTWARE PROCESS MODEL DESCRIPTION

I have chosen to implement the waterfall method for website development due to the following reasons:

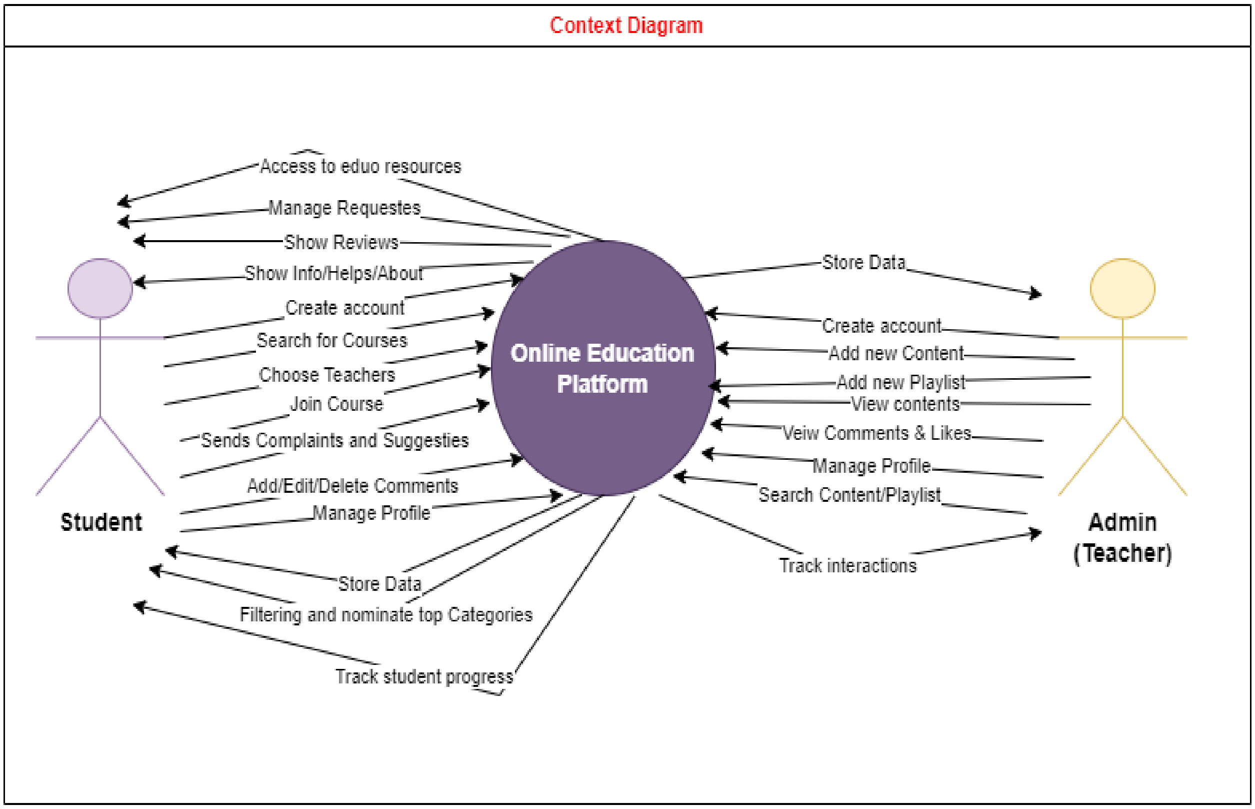
1. Some tools are still in the learning stage. It is not possible to engage in the agile experience because, despite its advantages, it is preferable to prioritize enhancing programming standards over relying on documentation and requirements.

2. The site's requirements are well-known, and the documentation is provided clearly and comprehensively.

3. The Spiral Model will be utilized to analyze and mitigate risks effectively, thereby yielding clear results in determining the requirements.

4. PRODUCT DEFINITION

Context Diagram



Click for High-Quality Context Diagram.

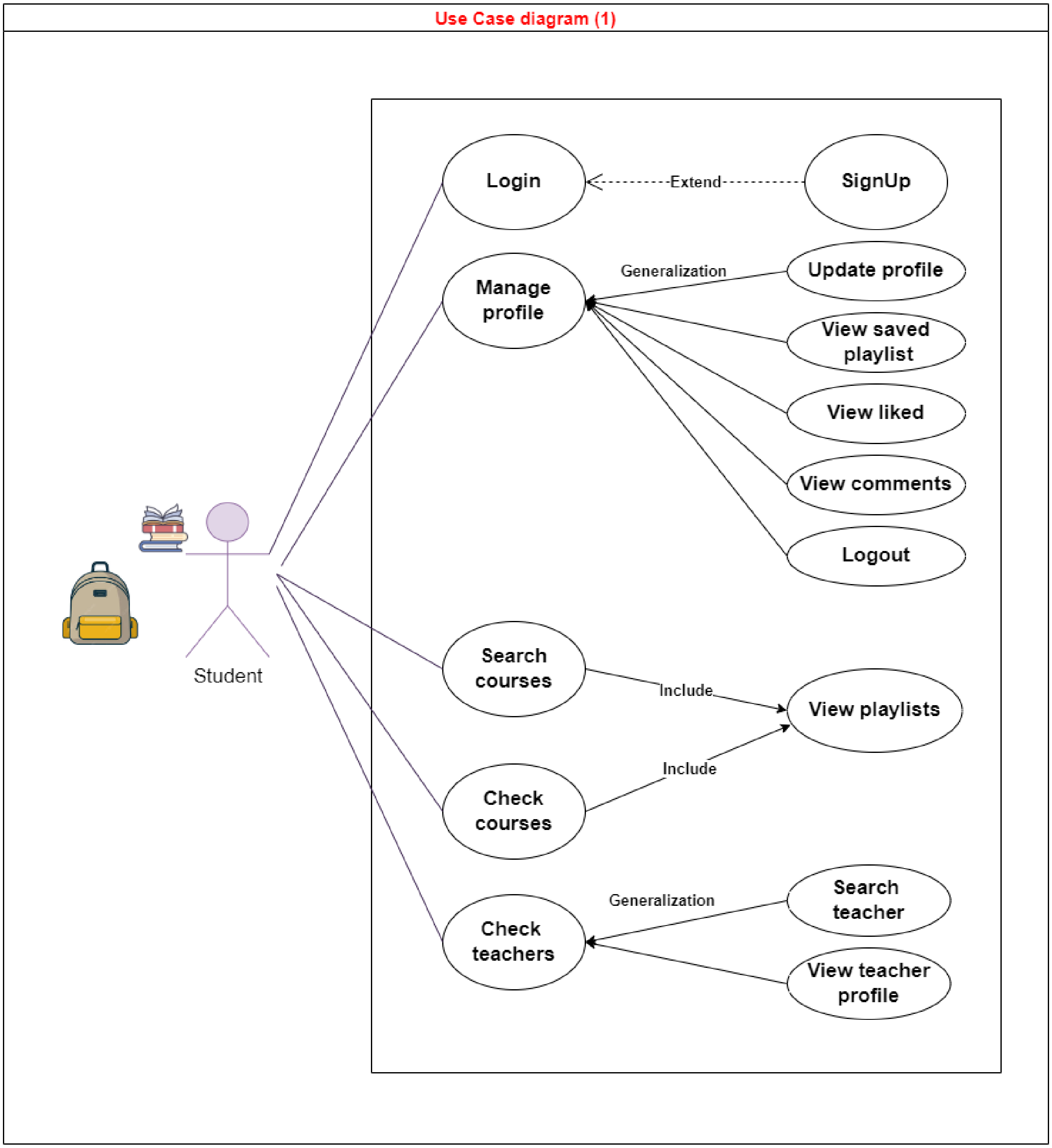
Student:

In this project, the student can search for the course they need with the teacher they prefer. They can do this by either specifying the course in advance or browsing through suggestions and making a selection. The teacher can be identified with the aid of various helpful suggestions, such as the number of likes received, ratings, comments, and other relevant information.

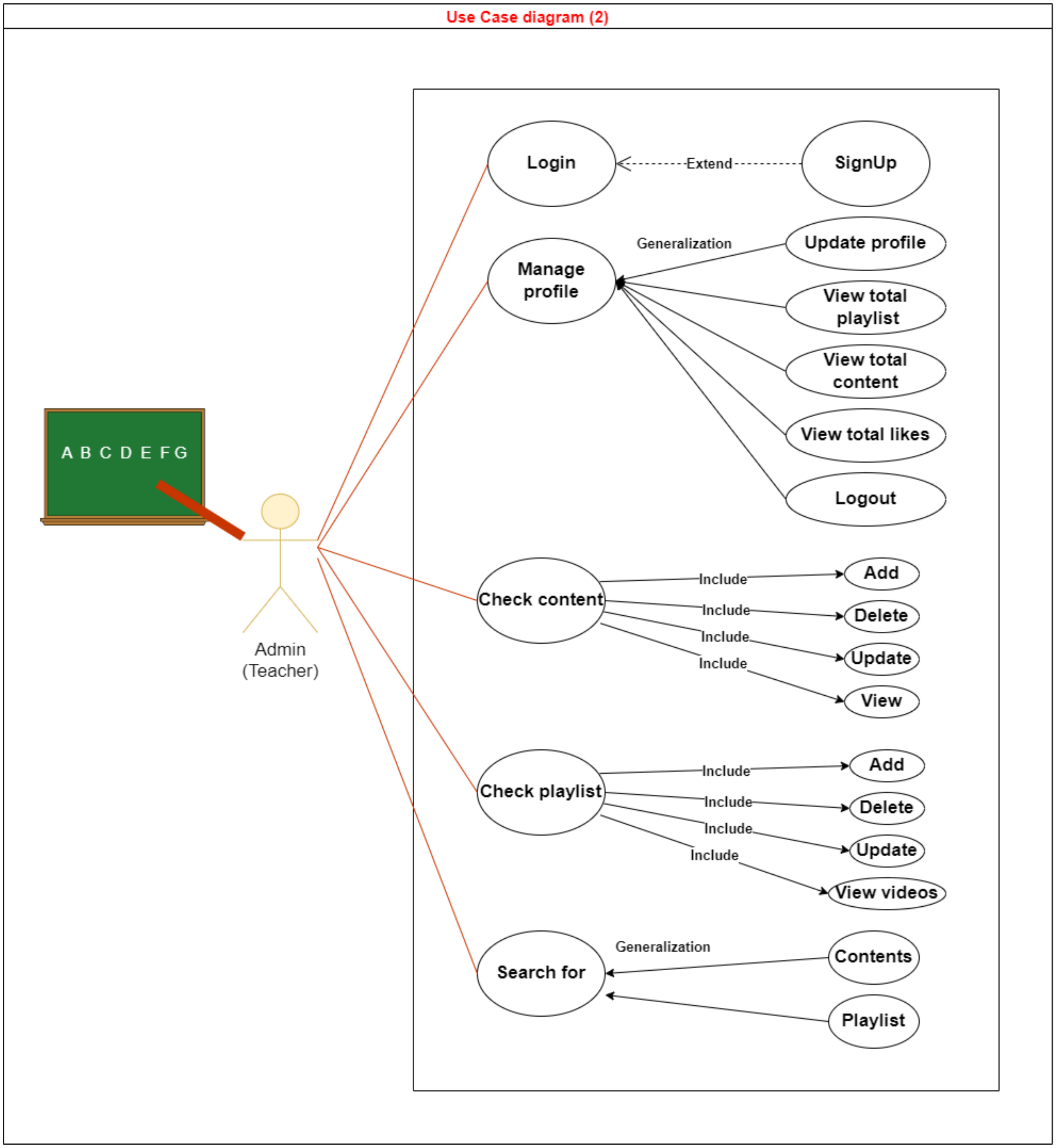
Admin (Teacher):

The teacher's responsibilities include organizing playlists, integrating content and videos with appropriate descriptions, monitoring comments and likes, and managing his permissions with the ability to delete, edit, and update as needed.

Use Cases:

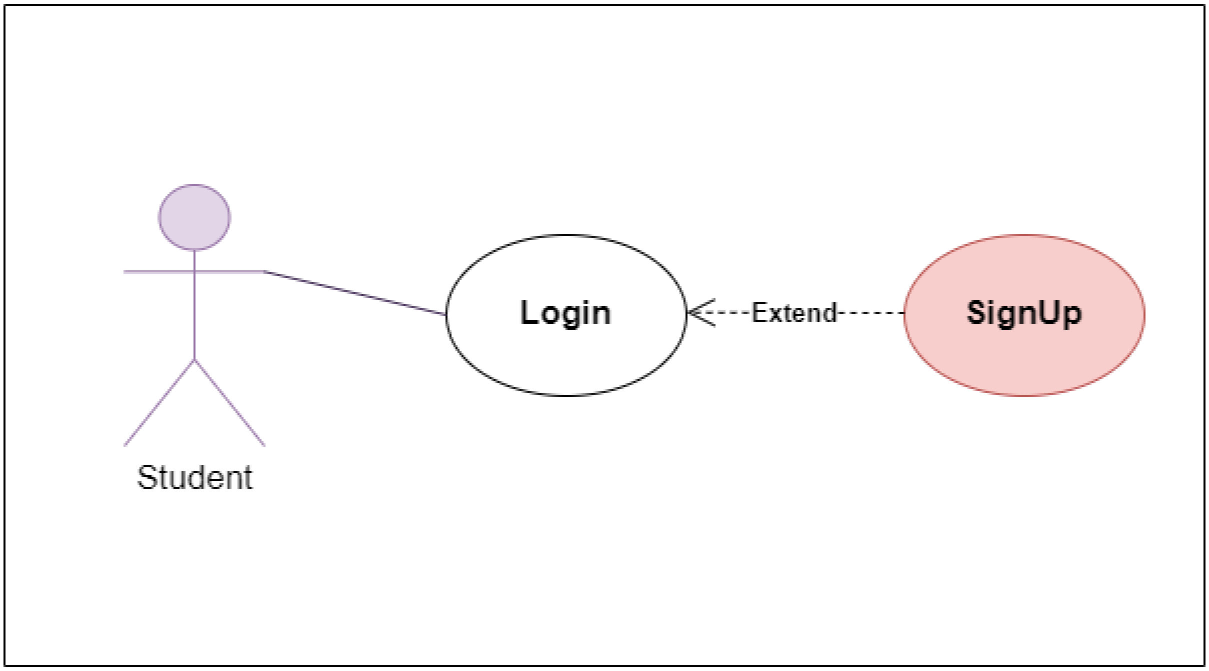


Click for High-Quality Use Case Diagram (1).



[Click for High-Quality Use Case Diagram (2).]

Use Case scenarios:



Use Case #1:

Name: Sign up

Participating Actor(s): Student

Entry Conditions: The student accesses the website and intends to perform actions without creating an account.

Exit Conditions: Account is created

Flow of Events:

1. Click on the Register button

2. Student enters the required information (name, email, password, confirm password, picture)

3. Student clicks the Register Now button

4. The student is directed to the home page.

Special Requirements:

1. If one of the required fields is left blank.

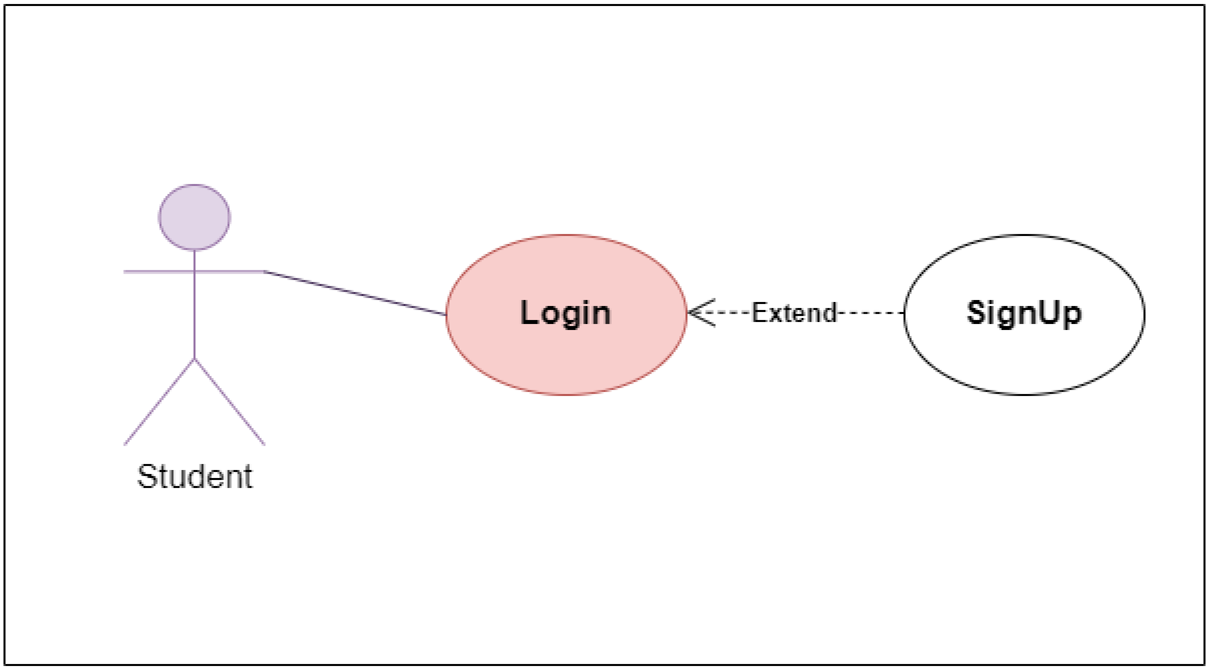
    - The system will display a warning message "Please fill out this field.".

2. If the password and confirmation do not match.

    - The system displays a message "Confirm password not matched!".

3. If the student's email is already in use.

    - The system displays the message "email already taken!".



Use Case #2:

Name: Login

Participating Actor(s): Student

Entry Conditions: The student, who has previously registered an account, seeks to utilize website features but finds that their session has expired (they are currently logged out).

Exit Conditions: The student logs into his account and begins using the processes available to him.

Flow of Events:

1- Student select the login button.

2- Enter the email and password in the provided fields.

3- Click on the "Login Now" button.

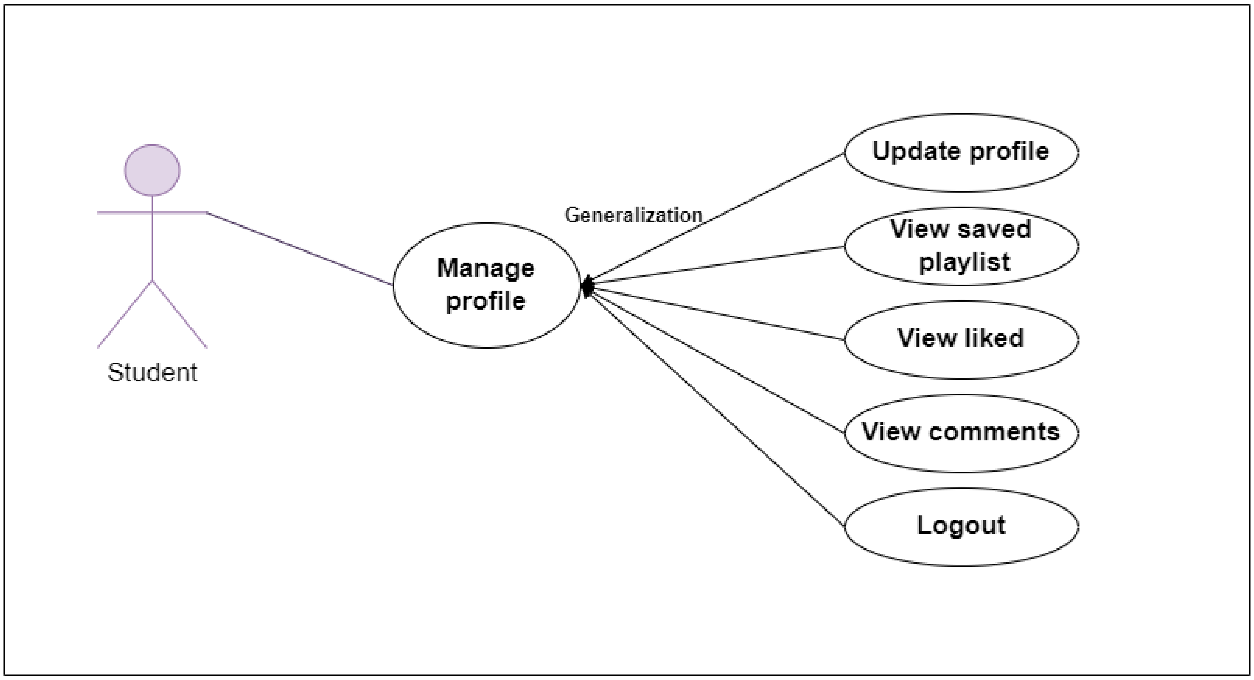
4- Upon successful authentication of the entered information with an existing user account.

5- Student is automatically logged in and redirected to the home page.

Special Requirements:

1. When one of the required fields is left blank.
   - The system will display an alert message "Please fill out this field".

2. If student enter any of the email or password field incorrectly.
   - The system will display a message "incorrect email or password!".



Use Case #3:

Name: Manage profile

Participating Actor(s): Student

Entry Conditions: The student must possess an account and be currently logged in.

Exit Conditions: The student's profile is modified in accordance with the requested alterations.

Flow of Events:

1. The student selects the profile icon- section.

2. The student clicks on the "View profile" button.

3- The system will display the "update profile" button.

4. The system will display at the bottom some information (No of saved playlists, no of liked tutorials, no of video comments).

5. The student clicks on the "Update profile" button.

6. The system displays the old information.

7. The student re-fill out the required field (email, picture, password, confirm password).

8. The system will validate the updated information and update it.

9. The system will display a saved successfully message for each field updated ("image updated successfully!", "password updated successfully!").

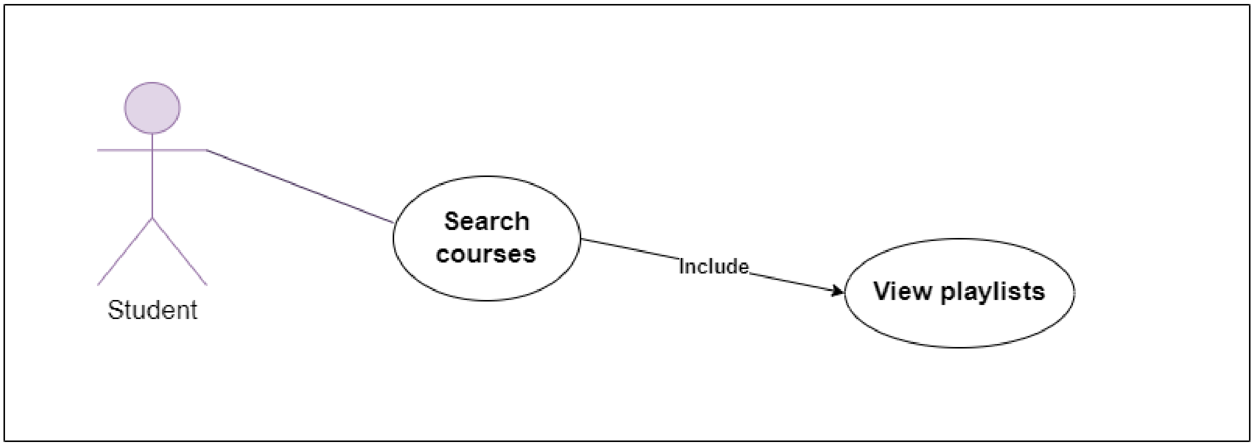
Special Requirements:

1- If the student wrote the previous password incorrectly.

- the system will display a message saying "Old password not matched! ".

2- If the new password does not match the Confirmed Password.

- the system will display a message "Confirm password not matched".



User Case #4:

Name: Search courses

Participating Actor(s): Student

Entry Conditions: The student has successfully signed in to the platform.

Exit Conditions: The student is successfully shown all the playlists associated with the name of the course he is searching for

Flow of Events:
1. The student clicks on the search bar.
2. The student enters the name of the course he wants
3. The student clicks on the search icon
4. The system checks whether this course name exists in the database
5. The system has verified the success of this session
6. The system displays all search results to the student from the accompanying playlists "View playlist" button.

7- The student clicks on the “View playlist” button.
8- The system displays a page with the playlist details (teacher name, specialty, and playlist description)
9- The system displays the videos that the teacher added to this playlist
10- The system displays the “Save” button.

- For saving playlist:
  - If the student clicks the “Save” button.
  - The system displays the message "Playlist saved!"

11- The student clicks on the video

12- The system displays a page containing the video, its title, date of publication, number of likes, and the teacher’s name and specialty.

13- The system displays the “View Playlist” button and the description below it

14- The system displays the “Like” button.

- For liking the video:
  - The student pressed the “Like” button.
  - System display message “Added to likes!”

14- At the end of the page, the system displays an interactive section for comments

15- The system displays a dedicated field for adding a comment, accompanied by an “Add Comment” button.

- For adding a comment:
  - The student writes his comment
  - The student clicks on the “Add Comment” button.
  - System display message “New comment added!”
  - System for adding student comments below the video
  - The system adds “Edit Comment” and “Delete Comment” buttons below the added comment

- For editing a comment
  - The student clicks on the “Edit Comment” button.
  - The system reloads the page and displays a section above the video called “Edit Comment”.
  - Text box display system with the old comment to change accompanied by two buttons "Cancel Edit" and "Update Now"
- To cancel the edit:
  - Student click on "Cancel Edit".
  - System reloads the page with the same comment that had been added before without editing.

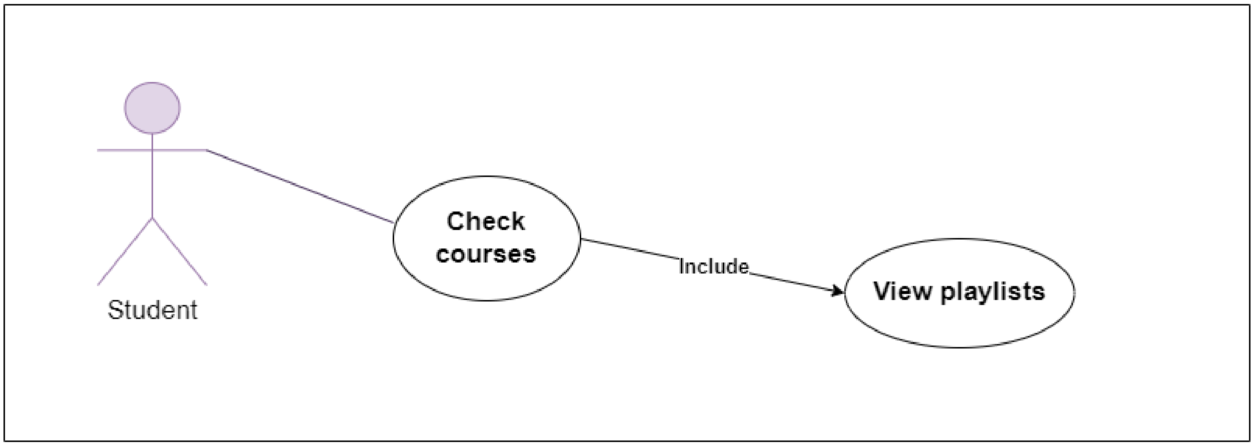
- To confirm editing:
    - Student changes the comment to the new one.
    - Student clicks on the "Update Now" button.
    - The system reloads the page with the updated comment.
    - The system displays the message "Comment edited successfully!".

- For deleting a comment:
    - Student click on "Delete comment"
    - The system displays the alert message “delete this comment?" to confirm deleting
        - if the student clicks "Cancel".
        - The system reloads the page without any change.
        - else if the student clicks "OK".
        - The system reloads the page by deleting the comment that is selected.
        - The system displays the message "comment deleted successfully!".

Special Requirements:

- If the course name is not found in the database.

- the system displays a message "No courses found!"



Use Case #5:

Name: Check courses

Participating Actor(s): Student

Entry Conditions: The student has successfully signed in to the platform.

Exit Conditions: All the different courses with different teachers are offered successfully

Flow of Events:

1. Use case (2).

2. The student selects the course icon from the side menu.

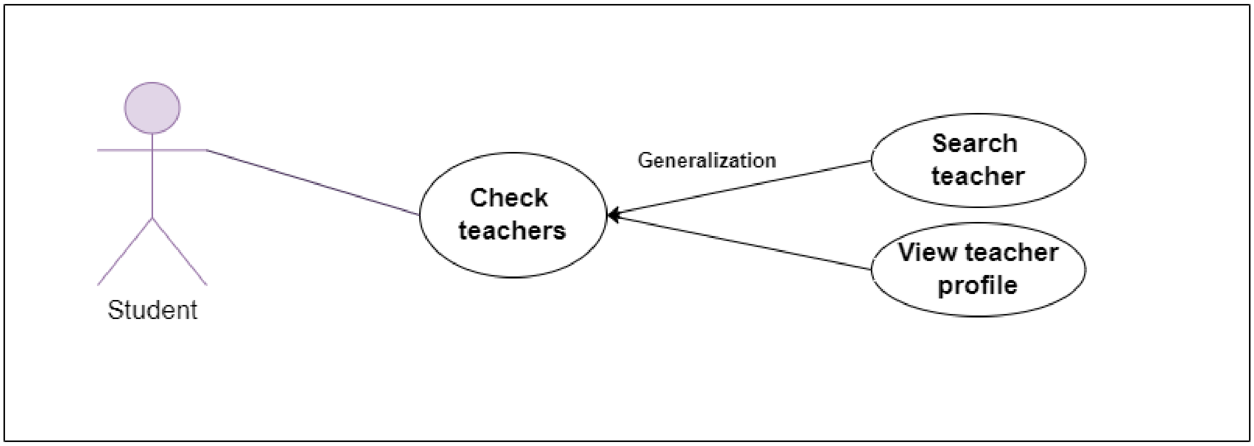
3. The system will show all the courses that teachers teach, accompanied by each playlist with the “View playlist” button.

4. Use Case (4) [7-15]

Special Requirements:

- If no courses are added.

- The system will display a message "No courses found!"



Use Case #6:

Name: Check teachers

Participating Actor(s): Student

Entry Conditions: The student has successfully signed in to the platform.

Exit Conditions: Each teacher's information has been successfully displayed.

Flow of Events:

1. Use case (2).
2. The student chooses the teacher icon from the side menu.
3. The system will display all teachers with simple information such as (No of playlists, no of total videos, no of total likes, no of total comments) accompanied by a “View profile” button.

4- The system will display a search bar for searching for the name of the teacher that the student wants to search for.

- Search among the already displayed on the page:
  - The student chooses the teacher by navigating and scrolling the page
  - Upon selection, student press "View profile"
- Search by the search bar:
  - The student enters the teacher’s name in the search bar.

  - The student clicks on the search icon.
  - The system verifies that the teacher’s name is present in the database.

- The result of the two-ways:
  - The system displays the teacher’s information on a separate full page, such as (the teacher’s name, specialty, no of playlists, no of total videos, no of total likes, no of total comments.).
  - The system displays all courses in the Playlist format created by the teacher with the "View playlist" button.

5- Use Case (4) [7-15].

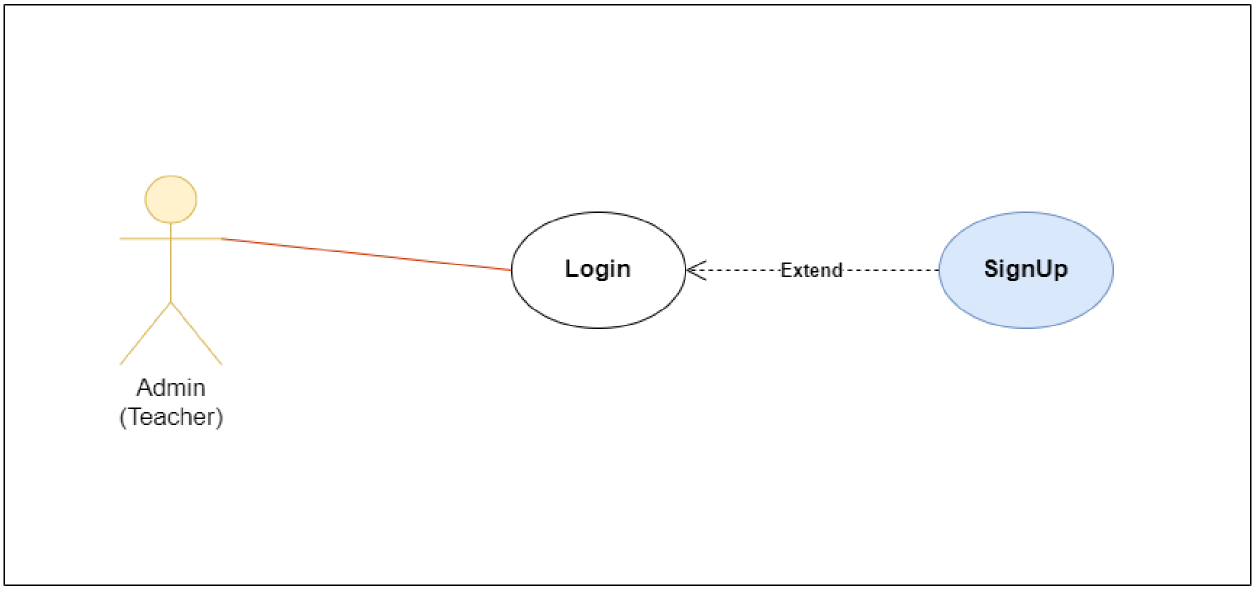
Special Requirements:

1- If the student enters the name of a teacher that is not present in the database.

  - The system will display a message “No results found!".

2- If the student presses the “Save” button again after saving the playlist.

  - The system will display a message saying "Playlist removed!".



Use Case #7:

Name: Sign up

Participating Actor(s): Teacher

Entry Conditions: The teacher accesses the website and intends to perform actions without creating an account.

Exit Conditions: Account is created

Flow of Events:

1. Click on the Register button

2. Teacher enters the required information (name, email, profession, password, confirm password, picture)

3. Teacher clicks the Register Now button

5. The system displays a message "New tutor registered! please login now".

6- The teacher clicks on the "login now" link which is located above the "Register Now" button.

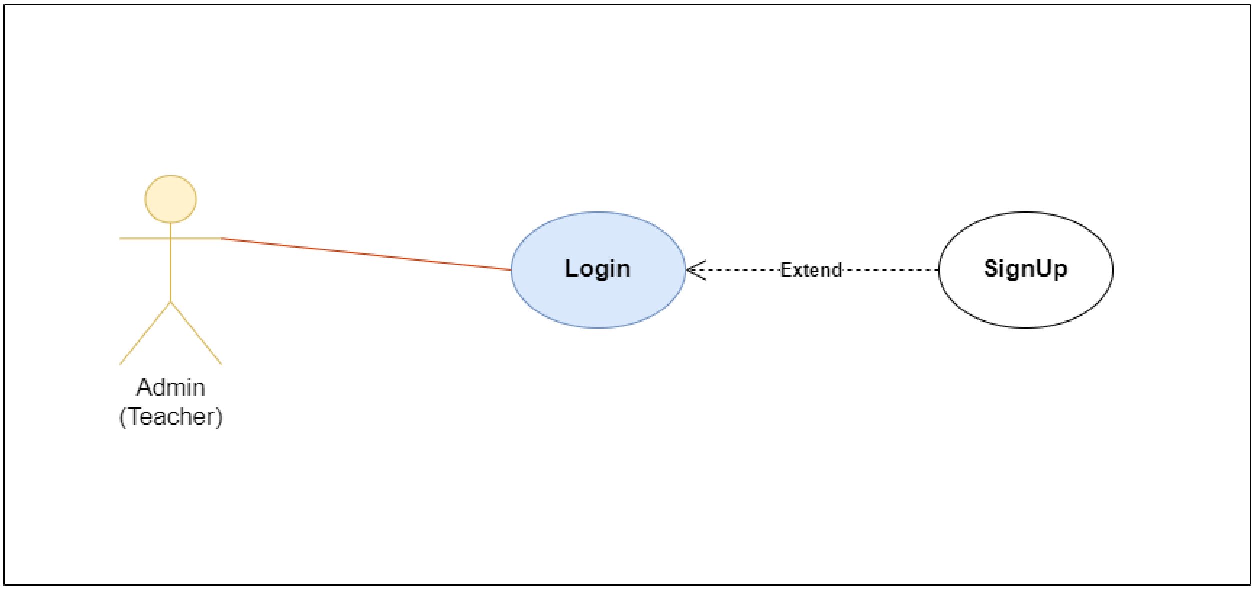
7- The teacher fills in the empty fields (email and password).

8- The teacher presses "Login Now".

9- The system turns the teacher into the home page.

Special Requirements:

1. If one of the required fields is left blank
    - The system will display a warning message "Please fill out this field.".
2. If the password and confirmation do not match.
    - The system displays a message "Confirm password not matched!".
3. If the teacher's email is already in use.
    - The system displays the message "email already taken!".
4. If the teacher enters the wrong email or password
    - The system will display the message “Incorrect email or password!”
5. If the teacher enters that Email already exists
    - The system will display a message “Email already taken!"



Use Case #8:

Name: Login

Participating Actor(s): Teacher

Entry Conditions: The Teacher, who has previously registered an account, seeks to utilize website features but finds that their session has expired (they are currently logged out).

Exit Conditions: The teacher logs into his account and begins using the processes available to him.

Flow of Events:

1- teacher select the login button.

2- Enter the email and password in the provided fields.

3- Click on the "Login Now" button.

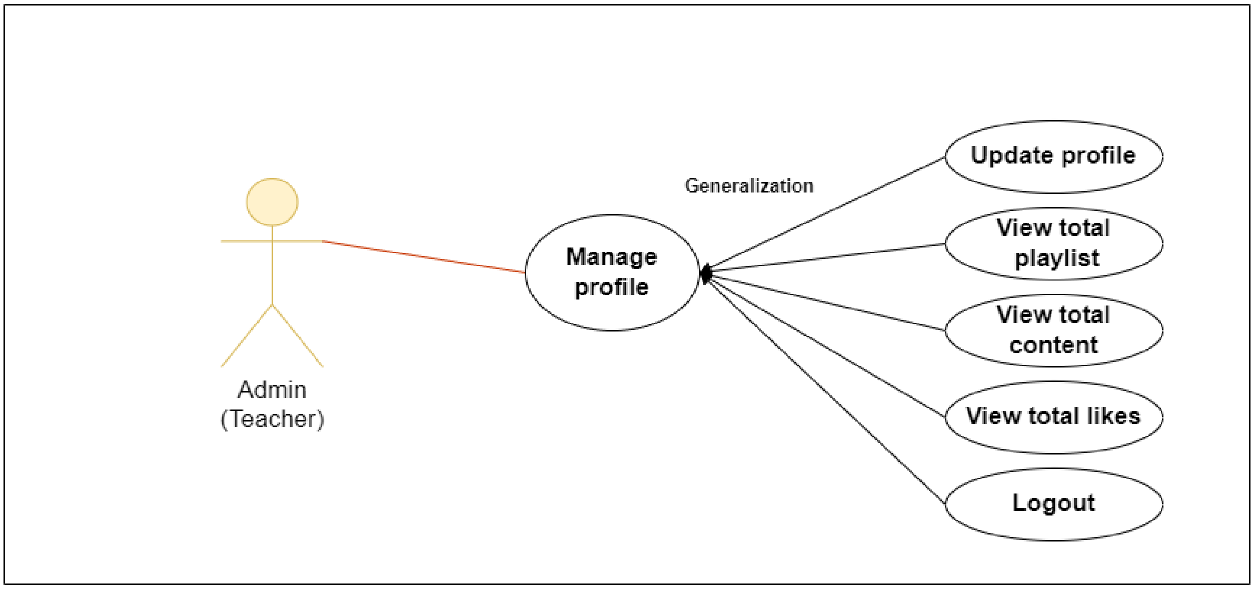
4- System check authentication of the entered information with an existing user account.

5- Teacher is automatically logged in and redirected to the home page.

Special Requirements:

1. When one of the required fields is left blank.
   - The system will display an alert message "Please fill out this field".

2. If teacher enter any of the email or password field incorrectly.
   - The system will display a message "incorrect email or password!".



Use Case #9:

Name: Manage profile

Participating Actor(s): Teacher

Entry Conditions: The Teacher must possess an account and be currently logged in.

Exit Conditions: The Teacher's profile is modified in accordance with the requested alterations.

Flow of Events:

1. The teacher selects the profile icon- section.

2. The teacher clicks on the "View profile" button.

3- The system will display the "update profile" button.

4. The system will display at the bottom some information (No of total playlists, no of total likes, no of total videos, no of total comments).

5. The teacher clicks on the "Update profile" button.

6. The system displays the old information.

7. The teacher re-fill out the required field (name, email, picture, profession, password, confirm password).

8. The system will validate the updated information and update it.

9. The system will display a saved successfully message for each field updated ("username updated successfully!", "profession updated successfully!"," email updated successfully!"," image updated successfully!", "password updated successfully!").

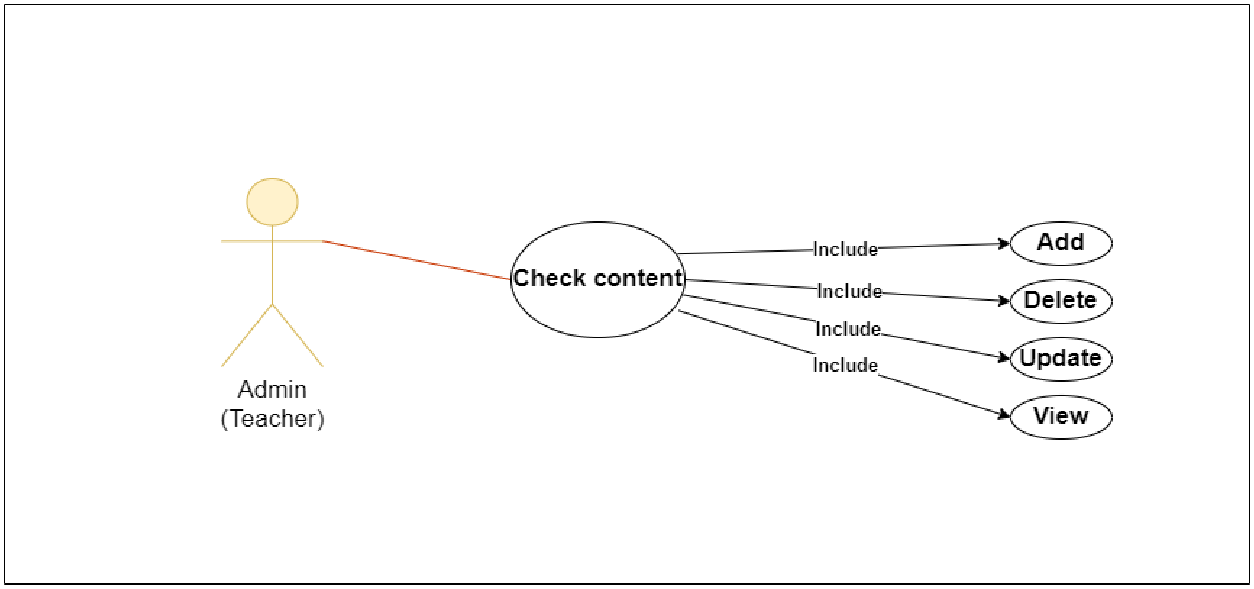
Special Requirements:

1- If the teacher wrote the previous password incorrectly.

- the system will display a message saying "Old password not matched! ".

2- If the new password does not match the Confirmed Password.

- the system will display a message "Confirm password not matched".



Use Case #10:

Name: Check content

Participating Actor(s): Teacher

Entry Conditions: The Teacher must possess an account and be currently logged in.

Exit Conditions: New course uploaded.

Flow of Events:

1- The teacher clicks on “Contents” in the side menu

2- The system will display the added contents

3- The system will show an “Add Content” button written above it

4- The teacher clicks on the “Add Content” button.

5- The system will show a page for downloading content with the information to be filled in, such as (video status (active, inactive), video title, video description, video playlist (from those previously added), select thumbnail, select video)

6- The teacher fills in the fields

7- The teacher clicks on the “Upload Video” button.

8- The system displays the message “New course loaded!”

9- The system reloads the page and shows the added course

10- The system appears under the added content buttons “Update,” “Delete,” and “View Content.”

- To update:

  - The teacher clicks the “Update” button.
  - The system displays the same information as the content that was previously created to be updated.
  - The teacher fills in the fields with the updated data.
  - The teacher clicks on “Update Content”.
  - The system displays the message “Content has been updated!”.

- To delete:

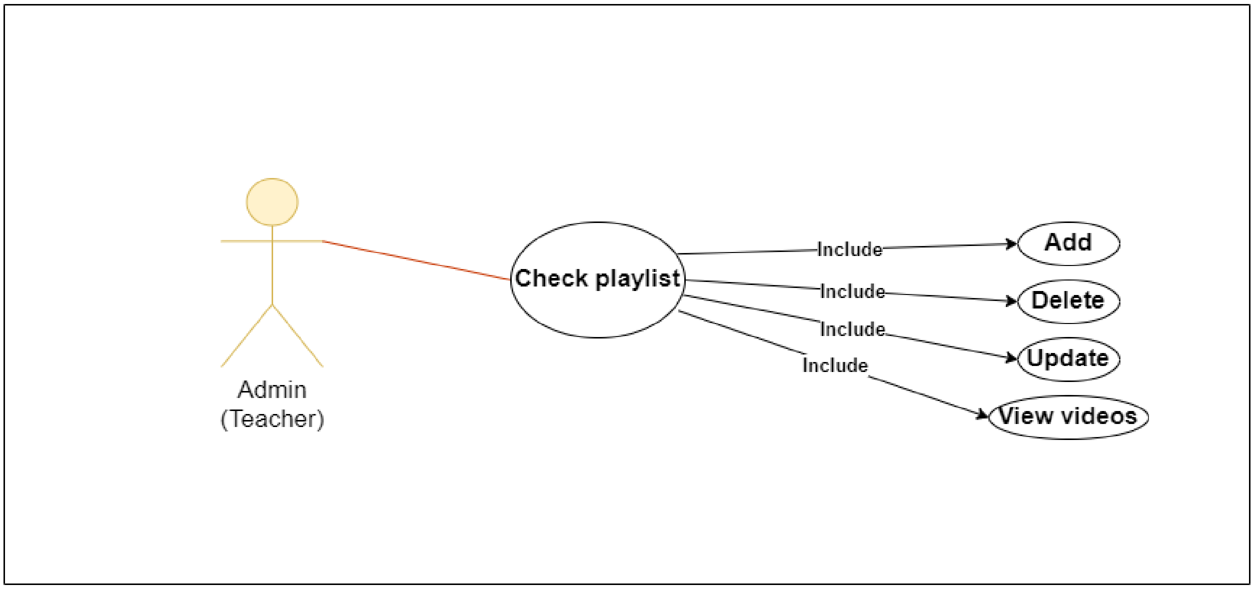
  - The teacher clicks “Delete”.
  - System confirmation message appears.
  - The system and content update page has been deleted.

- To display the content:

  - The teacher presses the "View content" button.
  - The system displays a page containing the content, the date it was published, its title, description, and the number of likes and comments.
  - The system displays two buttons, “Update” and “Delete.”
  - What happens in the update and delete step previously.
  - The system shows a “Delete comment” button for any comment that the teacher does not want.
  - The teacher presses the "Delete comment" button.
  - The system displays a confirmation message "Delete this comment?"
  - The system reloads the page with the changes.

Special Requirements:

1. When one of the required fields is left blank.
    - The system will display an alert message "Please fill out this field".



Use Case #11:

Name: Check playlist

Participating Actor(s): Teacher

Entry Conditions: The Teacher must possess an account and be currently logged in.

Exit Conditions: A new playlist has been created

Flow of Events:

1- The teacher clicks on "Playlist" located in the side menu of the page

2-The system will display a page containing the fields to be filled out

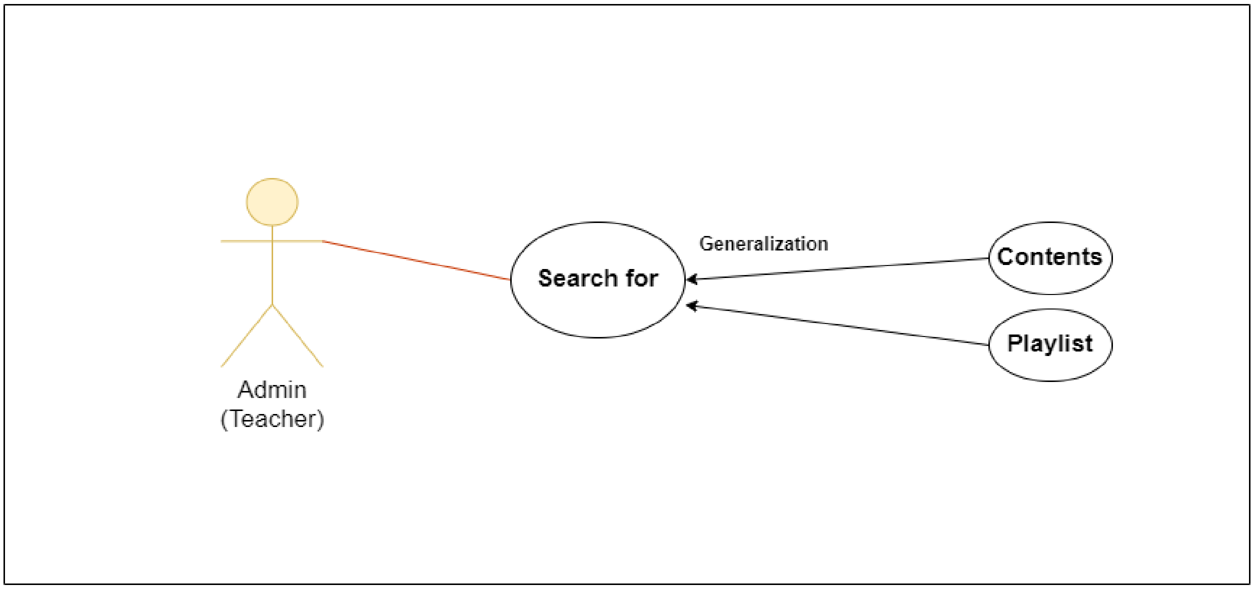
3-The teacher fills in the required fields such as (playlist status {active, deactive}, playlist title, playlist description, playlist thumbnail).

4- Teacher clicks "Create Playlist"

5- The system displays the message "A new playlist has been created!".

Special Requirements:

1. When one of the required fields is left blank.
    - The system will display an alert message "Please fill out this field".



Use Case #12:

Name: Search for

Participating Actor(s): Teacher

Entry Conditions: The Teacher must possess an account and be currently logged in.

Exit Conditions: Contents and playlists created matching the search name will appear successfully

Flow of Events:

1- The teacher presses the search bar

2- The teacher writes the name of the content or playlist he is looking for

3- The teacher clicks on the search icon

4- The system verifies the existence of what was written in the database

5- The system displays all results related to the search process, whether content or playlist

6- The system displays "Update", "Delete" and "View Content/Playlist" buttons.

7- Use cases 10 & 11

Special Requirements:

- If the content/playlist name is not found in the database.

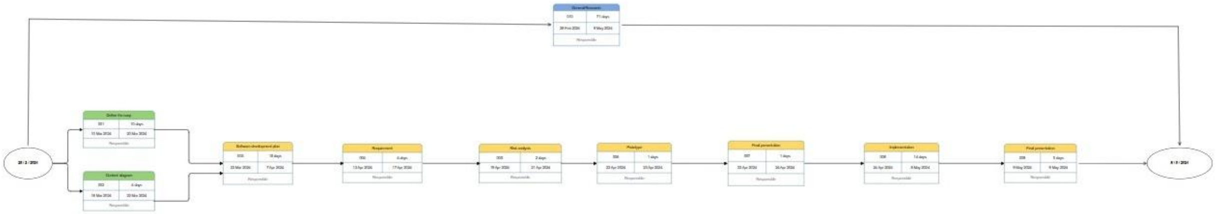
  - the system displays a message "no contents found!" or “no playlists found!” or both.

5. PROJECT ORGANIZATION

Matrix of Responsibilities

Team Members	Logina Mohamed
Concepts	
HTML	X
CSS	X
Java Script	X
PHP	X
My SQL	X
Security (hash tables)	X
Unit Testing	X
Research	X

6. PERT Chart



Click to High Quality Part Chart

7. VALIDATION PLAN

Test Strategy

Definition of Done:

- The web application fulfills all functional requirements outlined in the project scope.
- It boasts a user-friendly interface, intuitive design, and accessibility.
- All functionalities undergo comprehensive testing to ensure they are free from critical defects.
- The application is deployed and prepared for utilization by the project's deadline.

Success Criteria:

- The web application works properly and comfortably.
- Students can easily navigate through the website to choose a course and follow tutorials with the appropriate teacher.
- The categories represent the accuracy of going to the destination that the student wants and achieving his goal.

Main Test Plan:

1- Registration and Authentication for Students and Teachers:

- Assess the user registration process to confirm the successful creation of accounts.
- Validate authentication procedures including login and logout functionalities.

2- Find the right course:

- Testing the research process that the student will undertake.
- By writing a set of course names to determine whether all courses related to the name will be produced.

3- Access to the course:

- Testing the process of the student joining the course and his ability to interact with it

- Ensure that the student can save the video, interact, and make comments on it

4- Ability to choose the appropriate teacher:
   - Testing whether the student can search for specific teachers and choose the appropriate one for him.
   - By providing teachers' profiles so that students can view them.

5- All materials and sources are available:
   - Testing the process of adding the list and content from the teacher.
   - Ensure the success of the process of adding the tutorial and the possibility of modifying, updating, and deleting it.

6- Performance and Scalability Assessment:
   - Perform performance tests to verify the application's capability to manage anticipated user loads.
   - Evaluate scalability to ensure the application can expand to accommodate increases in user base and data volume.

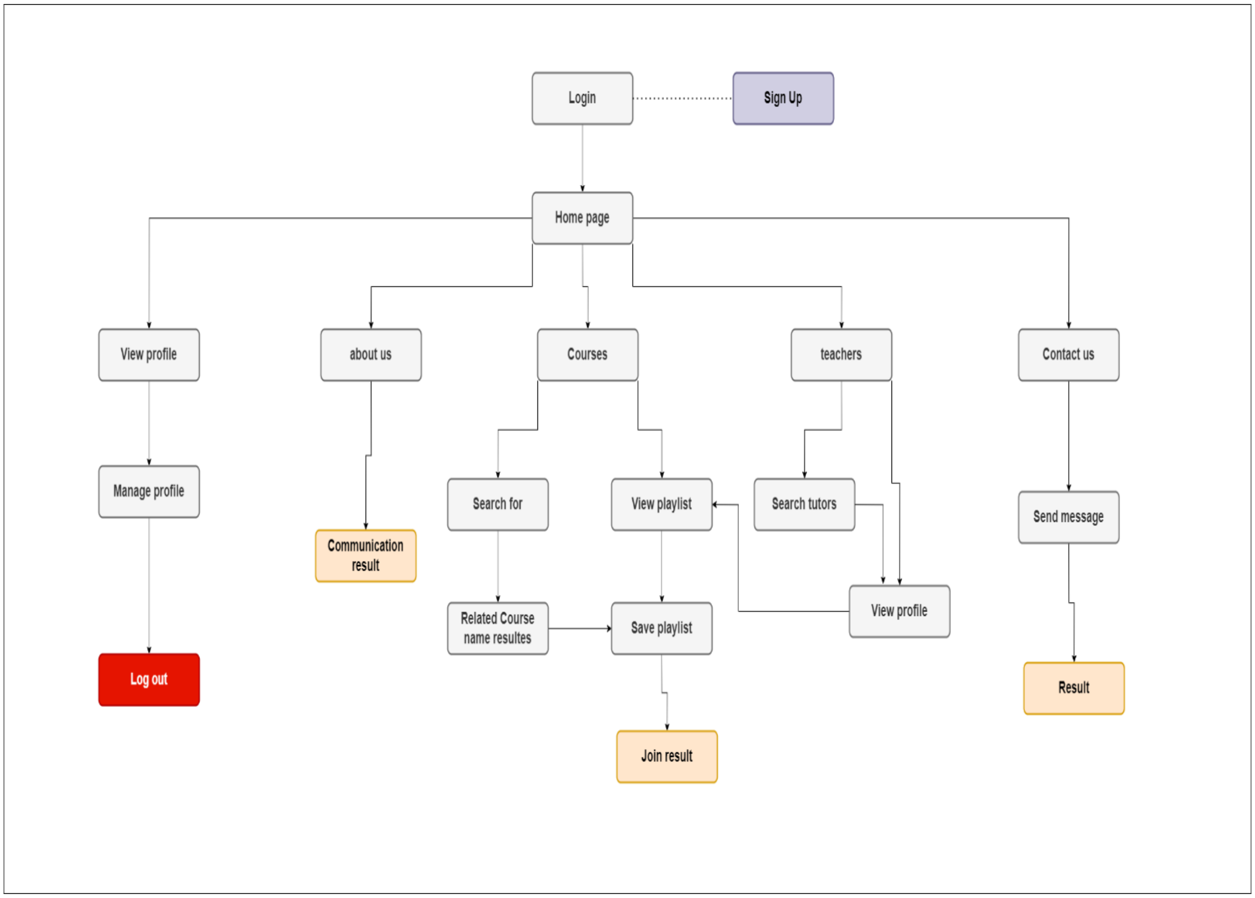
7- Testing for Accessibility and Usability:
   - Perform usability tests to collect feedback on user experience and interface design.

8- Testing for Security:
   - Confirm secure handling of sensitive user data to prevent unauthorized access.

9- Deployment and Environment Assessment:
   - Review the application's performance on different browsers and devices to guarantee compatibility and responsiveness.

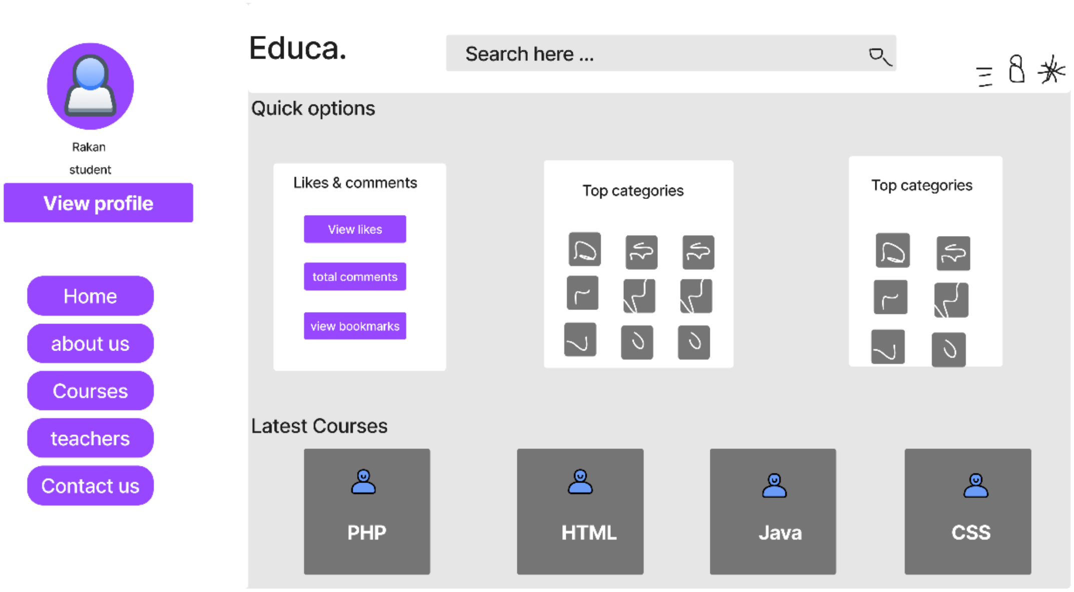
10- External users Testing:
   - Engage end users in testing to confirm that the application fulfills their needs and meets expectations.
   - Resolve any feedback or concerns raised during testing prior to finalizing the release.

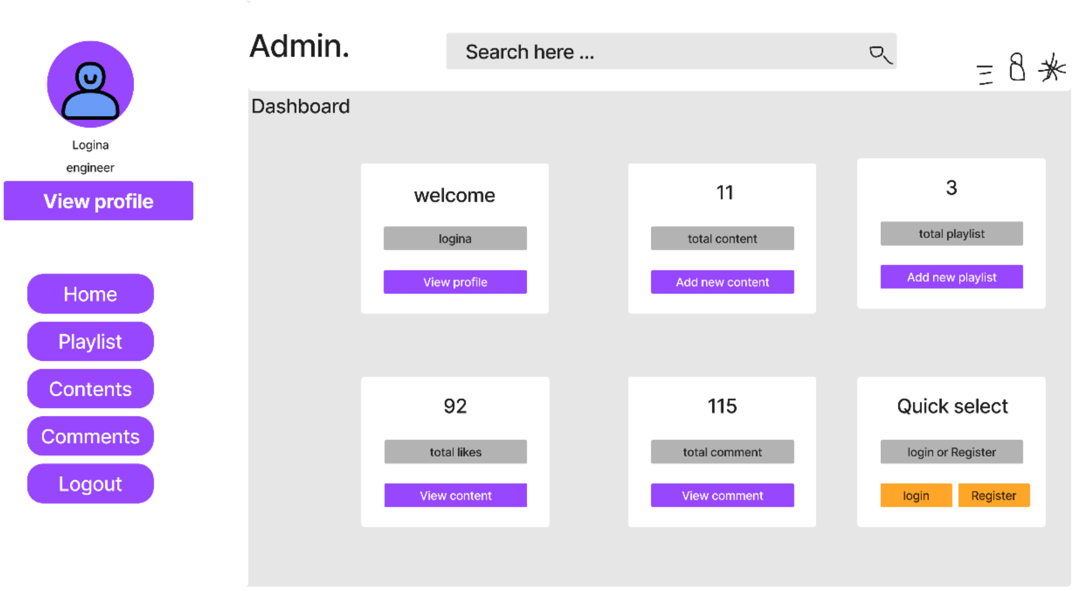
Wireframes:



Click for High-Quality Wireframes

Figma interfaces:





8. RISK ASSESSMENT

Risk Identification

The risk of unauthorized access to user information poses potential financial consequences, risks of personal information theft, misuse, and potential damage to the website's reputation.

Website downtime resulting from technical issues can render the site inaccessible to users, potentially undermining customer trust and impacting sales.

Technical vulnerabilities, including software vulnerabilities, pose a risk to system security and may lead to unauthorized access and data breaches.

Inappropriate content posting can diminish user satisfaction, tarnish the website's reputation, and result in legal repercussions.

Cyber-attacks, such as ransomware or phishing attempts, pose significant threats to website security and integrity.

Risk Prioritization

1- Cybersecurity breaches.
2- Interruptions in website availability.
3- Software weaknesses.
4- Posting of inappropriate content without authorization.

Risk Mitigation

1- Addressing Cybersecurity Breaches:

- Deploy strong cybersecurity protocols, such as firewalls, intrusion detection systems, and antivirus software.
- Ensure consistent software and system updates to address identified vulnerabilities.

2- Interruptions in website availability.:

- Employ redundant servers and backup systems to mitigate the effects of downtime.
- Conduct regular testing of backup and disaster recovery protocols to verify their efficacy.

3- Software weaknesses.:

- Maintain up-to-date software and systems to address any identified issues.
- Conduct regular security checks and promptly resolve any vulnerabilities to ensure system integrity.

4- Posting of inappropriate content without authorization:

- Offer students reporting tools for inappropriate content.
- Regularly monitor content generated by teachers to detect and rectify any violations.

9. CONFIGURATION AND VERSION CONTROL

For the app: X.Y.Z

X = Major Releases

Y = Minor Feature Updates/Cosmetic Updates

Z = Bug Fixes

Example: First major release, second bug fix: Version 1.2

Any changes made to the project will be included in the GitHub.

10. TOOLS

- **My SQL (PHPAdmin)**: To establish and oversee database operations.
- **Visual Studio Code:** To compose and execute commands: -
  - HTML
  - CSS
  - Java Script
  - PHP

11. ARCHITECTURE

- PC or laptops for programming and testing