EventHub

Use Case diagram & Description

PRESENTED BY

MEMBER 1: AHMAD MASOOD (LEADER)

MEMBER 2: M. AHMAD HASSAN

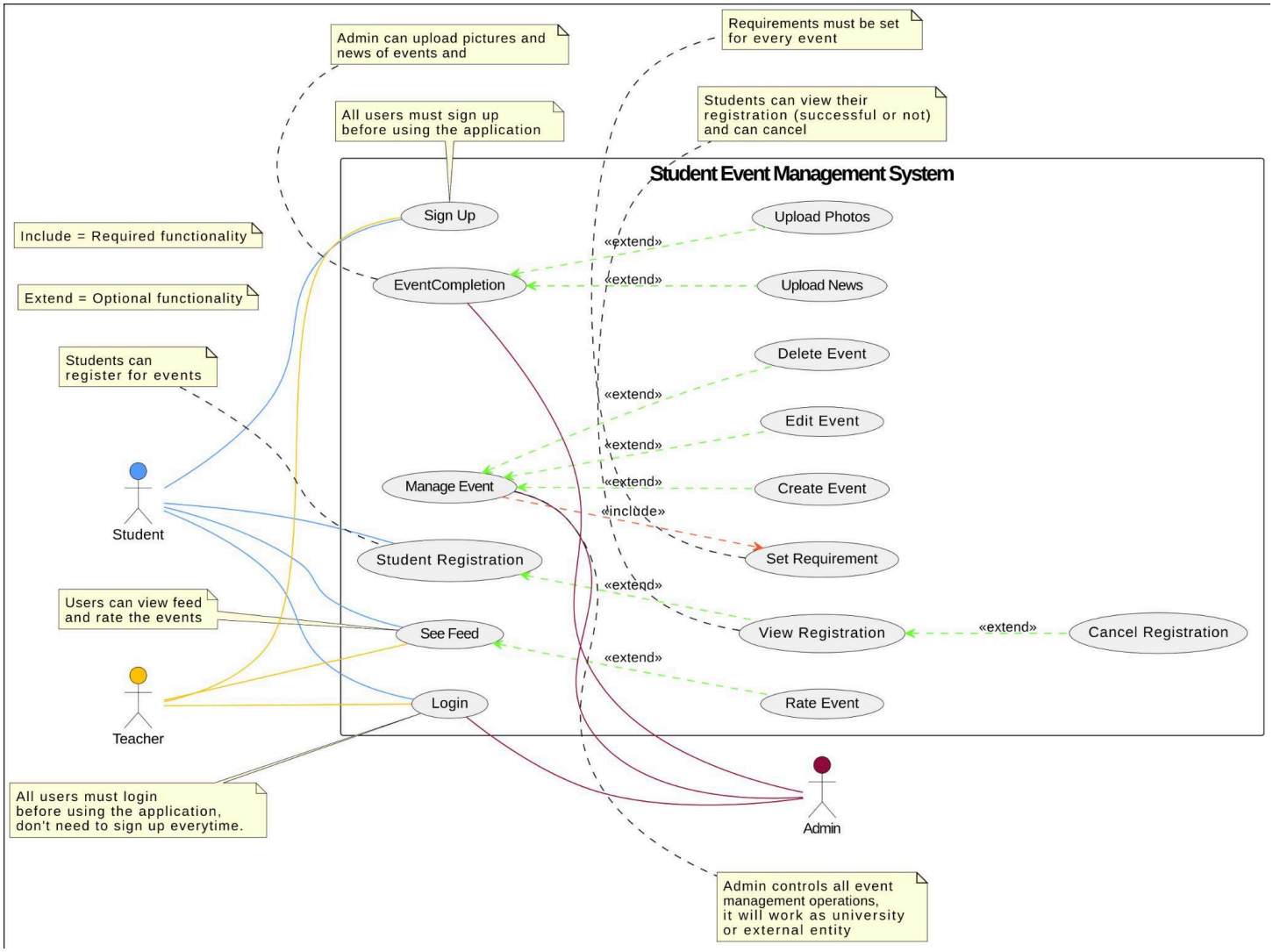
MEMBER 3: M. WALEED

MEMBER 4: TALHA WAINS

SUPERVISED BY

Dr. ONAIZA MAQBOOL

Use Case Diagram:



BY: M. Waleed

UC 1 : Event Completion

Primary Actor:
Admin (Event Organizer)

Stakeholders and Interests:
- Student:
- Wants to see event outcomes, news, and photos, and provide feedback..

- Department:
- Wants a record of completed events for documentation and future planning.

- Event Organizer (Admin):
- Wants to officially mark the event as completed and share updates.

Preconditions:
- The event must have taken place.
- The admin must be login in the system.

Success Guarantee (postconditions):
- The event status is updated to "Completed."
- Photos and news are uploaded for students and teachers to view.
- The system allows students and teachers to provide ratings and feedback.

Main Success Scenario (or Basic Flow):

- The admin logs into the system.
- The admin navigates to the event management section.
- The admin selects the complete event.
- The system prompts post-event details (summary, key highlights, photos).
- The admin uploads event news, images, and key outcomes.
- The system updates the event status to "Completed."
- The system enables the rating and feedback feature for students and teachers.

Alternate Scenario (or Extensions):

a. At any time, the System fails:
To ensure recovery of the student's data, the software saves the existing data of the company's post job and then when Company reopen the app, App will resume its working where Company had left so that they can recover from any step.

4a. The admin does not have all the post-event details:
- The system allows saving progress and updating later.

5a. Upload fails due to file size or format issues:
- The system notifies the admin and prompts for a compatible file.

Special Requirements:
- The App should be developed in Java.
- The student registration should be easily visible to the users of the platform by using font size that is easily readable like 12 font size.

Frequency of Occurrence:
It could be nearly continuous.

Open Issues:
Open for feedback to improve use case text.

BY: M. Ahmad Hassan

UC 2: Student Registration

Primary Actor:
- Student

Stakeholders and Interests:
- Student:
- Wants to register easily for events they are eligible for and view their registration details.

- Department:
- Wants an organized system to manage student participation.

- Event Organizer:
- Wants to monitor student registrations and ensure participant limits are maintained.

Preconditions:
- The student must be login in the system.
- The event must be open for registration.

Success Guarantee (postconditions):
- The student is successfully registered for the event.
- The system updates the event's participant list.

Main Success Scenario (or Basic Flow):

- The student logs into the system.
- The student navigates the events section and selects an event.
- The system displays event details and eligibility criteria.
- The student confirms eligibility and proceeds with registration.
- The system registers the student and updates the participant list.
- The student receives confirmation of successful registration.

Alternate Scenario (or Extensions):

***a. At any time, the System fails:**

To ensure recovery of the student registration data, the software saves the existing data of the company's post job and then when Company reopen the app, App will resume its working where Company had left so that they can recover from any step.

1a. Student is not eligible for the event:
- The system displays error messages and does not allow registration.

5a. Event registration limit is reached:
- The system notifies the student that registration is full and prevents further registration.

Special Requirements:
- The App should be developed in Java.
- The student registration should be easily visible to the users of the platform by using font size that is easily readable like 12 font size.

Frequency of Occurrence:
It could be nearly continuous.

Open Issues:
Open for feedback to improve use case text.

By: Ahmad Masood

Use Case 3: Manage Event

Primary Actors:

- ☐ Students
- ☐ Teachers

Stakeholders and Their Interests:

Students:

- Want to create and manage events they are hosting.
- Need an easy way to edit event details if changes are needed.
- Want the ability to delete an event if it is no longer happening.

Teachers:

- Want to create academic or extracurricular events.
- Need to ensure event details are correct and up-to-date.
- Require control over event modifications or cancellations.

Description:

This use case defines the process of managing events in the system. Users can create, edit, and delete events as per their requirements. Additionally, they can set event-related details such as date, time, and location.

Preconditions:

- The user must be logged into the system.
- The user must have permission to create, edit, or delete events.

Main Flow:

1. The user navigates to the "Manage Event" section.
2. The system displays options for managing events:

Create Event:

1. The user selects "Create Event."

2. The system prompts for event details (name,description, location, etc.).

3. The user enters the details and submits the form.

4. The system saves the event and adds it to the event list.

Edit Event:

1. The user selects an existing event to edit.

2. The system displays the event's details.

3. The user modifies the necessary fields and submits the changes.

4. The system updates the event information.

Delete Event:

1. The user selects an event to delete.

2. The system prompts for confirmation.

3. Upon confirmation, the event is removed from the system.

Set Event Requirements:

1. The user selects "Set Event Requirements."

2. The system displays available requirement fields (e.g., prerequisites, maximum participants, required documents).

3. The user enters or updates the requirements.

4. The system saves the event requirements.

Alternative Flows:

Invalid Event Details:

If required fields are missing during creation or editing, the system prompts the user to complete them.

Unauthorized Deletion:

If a user without proper permissions attempts to delete an event, the system denies access.

Postconditions:

The event is successfully created, updated, or deleted as requested.

By Talha Arif Wains

Use case 4:

Title:

"**See Feed**" and "**Event Rating**" (extended)

Actors:

- Student
- Teacher

Stakeholders:

Students & Teachers:

They want to stay informed about recent event updates and able to provide their feedback through ratings.

Admin:

Ensures the feed is populated by uploading news and images from events, enabling users to view content and rate events.

Pre-conditions:

The user must be authenticated in the system.

The Admin has successfully uploaded event-related news or photos so that content is available for display in the feed.

Post-conditions:

The user successfully views a feed containing current event updates (news/photos).

Any rating submitted by the user is saved.

Special Requirements:

Only authenticated users may access the feed and use the rating feature.

The feed will only display events that include content uploaded by the Admin.

Frequency of Occurrence:

This process is expected to occur frequently, many times a day, as users routinely check for updates and provide feedback on events.

Main Success Scenario:

- A user logs into the system with valid credentials.
- The user navigates to the "See Feed" section.
- The system confirms that there are event updates (news/photos) available.
- The feed is displayed, showing event content along with an accessible option to rate event.
- The user submits a rating (e.g. star system or emojis).
- The system stores the rating, updating the event's overall rating accordingly.
- The new rating, are visible in the feed.

Alternate Scenario:

If the Admin has not yet provided any event updates, the feed will show a display message such as "No event updates available!" and the option to rate will not be activated.

Failed Scenario:

If the user is not logged in, the system will block access to the feed and rating functions, redirecting the user to the login page.

Should a technical error occur (for example, a problem with the database fetch), the feed might fail to load or the rating might not be recorded, prompting an error message with proper message for the user.
