

Software Requirement Specification (SRS)

for Student Clubs Event Management Platform

Problem Background

The primary purpose of this platform is to enable student club coordinators to post requests for scheduling events in an academic institute, which would be subject to approval by an events coordinator. Post this approval, the event would be added to the platform for registration by students. The platform would also have dedicated pages for the clubs, which the respective club's coordinator would manage.

Stakeholders / Users

The primary users of the platform are (i) students of the university, (ii) the coordinators of various clubs, (iii) an Admin for the platform.

Functional Requirements

The use cases for the different user groups are listed below. Details of the use cases are given later. For each use case, its priority is also mentioned.

Name of the Use Case	Purpose	Precondition
For Students		
Explore Upcoming Events (essential)	For users to explore upcoming events organised by various clubs	Student logged in
Explore Clubs (essential)	For users to explore and obtain information about the various registered clubs.	Student logged in
Log in (essential)	To log in the platform	None
Register for Events (essential)	To register for events	Student logged in
View the Profile (desirable)	Allow students to view and edit their profile	Student logged in
Propose a new club (essential)	Allow students to propose a new club	Student logged in
For Club Coordinator		
Propose an Event (essential)	To propose a new event	Club coordinator logged in
Edit an event's information (essential)	Edit the event's information	Club coordinator logged in
For Admin		
Approve an event (essential)	Approve an event	Event has been created; coordinator logged
Approve a new club (essential)	Events coordinator	Coordinator logged in
Change the events coordinator (optional)	Change the coordinator or his/her info	Coordinator is logged in

Performance Requirements

1. The system shall be designed to support up to 1000 concurrent users without degradation in performance, ensuring a response time of less than 300 milliseconds for 95% of requests under this load.

2. The platform should be responsive and have an average response time of 300 milliseconds or less under normal load (of up to 1000 users).

Design Constraints

1. For user authentication institute's domain gmail address should be used.
2. The tech stack for the platform should be free and open source.

External Interfaces

1. The platform should be compatible with all major browsers like Safari, Chrome, Firefox, Brave, Microsoft Edge, etc.
2. The platform should use users' institute email address for login and authentication.
3. The platform should be able to send email notifications to students and other users (Optional).

Security / Privacy Requirements

1. Only authorised users should be able to execute the different use cases; the landing page is visible to all.
2. The platform shall implement security measures including but not limited to input validation and output encoding (Optional).

Use Case Details

1. Explore Upcoming Events

- 1.1. **Actor:** Students
- 1.2. **Purpose:** For users to explore upcoming events organized by various clubs.
- 1.3. **Event Flow:**
 - 1.3.1. User lands on the homepage of the website.
 - 1.3.2. User views the list of upcoming events displayed on the main landing page.
 - 1.3.3. If desired, the user can click on the "Home" button to refresh or return to the main landing page showcasing upcoming events.
- 1.4. **Special/Exceptional Requirements:**
None for this use case.

2. Explore Clubs

- 2.1. **Actor:** Students
- 2.2. **Purpose:** For users to explore and obtain information about the various registered clubs.
- 2.3. **Precondition:** Student is logged in
- 2.4. **Event Flow:**
 - 2.4.1. User clicks on the "Clubs" button on the navigation bar.
 - 2.4.2. User is redirected to the clubs' information page.
 - 2.4.3. On this page, for each club, the user can view the club information.

2.5. Special/Exceptional Requirements:

None for this use case.

3. User Log-In

3.1. Actor: Students, College Admin, Club Coordinators, Events Coordinator

3.2. Purpose: For students to log in to the platform/

3.3. Event Flow:

3.3.1. User navigates to the main landing page of the portal.

3.3.2. User clicks on the login button.

3.3.3. User is redirected to log in through the Google API.

3.3.4. After successful login, the user is redirected back to the main page; the login button is replaced with the logout button

3.4. Special/Exceptional Requirements:

None (google API will take care of incorrect logins)

4. Register for Events

4.1. Actor: Students

4.2. Purpose: To allow users to register for upcoming events.

4.3. Precondition: Student is logged in

4.4. Event Flow:

4.4.1. User explores upcoming events from the main landing page.

4.4.2. User clicks on an event to view its details.

4.4.3. User clicks on the "Register" button.

4.4.4. The user is registered and redirected to the homepage.

4.5. Special/Exceptional Requirements:

4.5.1. If not logged in, the user is prompted to log in.

5. View Profile

5.1. Actor: Students, Admin, Club Coordinators

5.2. Purpose: To allow users to view their personal profile, which displays their details and the events they've registered for.

5.3. Event Flow:

5.3.1. From the main landing page, user clicks on the "Profile" button in the navigation bar.

5.3.2. Users, if logged-in, are redirected to their respective profile page.

5.3.3. The user sees their name, roll number, email id, and profile type badge.

5.3.4. The user can view the events they've previously registered for.

5.4. Special/Exceptional Requirements:

5.4.1. If not logged in, the user is prompted to log in.

5.4.2. Profile type badges have specific classifications:

5.4.2.1. Student: Regular students.

5.4.2.2. Club Head: Club coordinators/heads for clubs.

5.4.2.3. Admin: Administrative personnel.

6. Propose an Event

6.1. **Actor:** Club Coordinators

6.2. **Purpose:** To allow club coordinators to propose the organization of events.

6.3. **Event Flow:**

6.3.1. User lands on the homepage and clicks on the "Clubs" option in the navigation bar, redirecting to the clubs' information page to locate and select their respective club from the list.

6.3.2. On the club-specific page, the user clicks on the 'Propose Event' button post which a form appears.

6.3.3. After filling out this form, the user clicks the "Submit" button.

6.3.4. The event proposal is sent for approval, and the user is redirected back to the club information page.

6.4. **Special/Exceptional Requirements:**

None for this use case.

7. Edit Clubs' Information

7.1. **Actor:** Club Coordinators, event coordinator, Admin

7.2. **Purpose:** To allow specific roles to edit the information of the clubs.

7.3. **Precondition:** Club coordinator logged in

7.4. **Event Flow:**

7.4.1. **For a Club Coordinator:**

7.4.1.1. Club Coordinator/Head selects their club from the "Clubs" page.

7.4.1.2. Clicks the 'Edit Club Information' button on the club-specific page.

7.4.1.3. Edits the club description and members list.

7.4.1.4. Clicks the "Save" button.

7.4.2. **For the Admin:**

7.4.2.1. Chooses the 'Manage' option from the navigation bar.

7.4.2.2. Selects the 'Edit Club Information' from management options.

7.4.2.3. Chooses the desired club and edits relevant fields.

7.4.2.4. Clicks the "Save" button.

7.5. **Special/Exceptional Requirements:**

7.5.1. Only Club Coordinator/Head can see the club-specific 'Edit' button.

7.5.2. The Events Coordinator and Admin access editing through the 'Manage' option.

8. Approve Event

8.1. **Actor:** Admin

8.2. **Purpose:** To evaluate and approve or reject event proposals from club coordinators/heads.

8.3. **Event Flow:**

8.3.1. Admin selects the 'Manage' option after logging in.

8.3.2. Chooses 'Pending Event Requests' from the available options.

8.3.3. Views a list of pending proposals and selects one to review its details.

8.3.4. Decides to 'Approve' or 'Reject' the event and clicks the relevant button

8.4. Special/Exceptional Requirements:

None for this use case.

9. Change the Events Coordinator

9.1. Actor: Admin

9.2. Purpose: To change the details and assign the role of Events Coordinator to a new individual.

9.3. Event Flow:

9.3.1. Admin logs in and selects the 'Manage' option from the navigation bar and chooses the 'Update events Coordinator' option.

9.3.2. Views current Student Coordinator's details and clicks on "edit" and updates the required details.

9.3.3. Saves the changes, updating access and profile badges for the current and new coordinators.

9.4. Special/Exceptional Requirements:

9.4.1. The events Coordinator cannot see or access this specific management option.

10. Request the Formation of a New Club

10.1. Actor: Students

10.2. Purpose: To propose the formation of a new club.

10.3. Event Flow:

10.3.1. User selects the 'Clubs' option on the homepage and lands on the clubs' information page.

10.3.2. Clicks on 'Propose a New Club' and fills out the form.

10.3.3. Submits the form, sending the request forward for approval.

10.4. Special/Exceptional Requirements:

None for this use case.

11. Approve New Club Requests

11.1. Actor: Admin

11.2. Purpose: To evaluate and approve/reject new club formation proposals.

11.3. Event Flow:

11.3.1. User logs in, selects 'Manage', and navigates to 'Pending New Club Requests'.

11.3.2. Reviews and decides to 'Approve' or 'Reject' a specific club proposal. Approved proposals are made available for Admin review.

11.3.3. Admin reviews and finalises the decision by either approving, thereby adding the new club to the portal, or rejecting.

11.4. Special/Exceptional Requirements:

11.4.1. Only proposals approved by the Events Coordinator are visible to the Admin.

Functional Requirements in terms of Features/Functionality.

Functional requirements can also be stated directly as functionality or features the application should support. This section specifies the functionality in terms of features.

1. The system shall allow users with an active domain Gmail ID of the institute for various functionalities. The authentication process shall verify the domain of the Gmail ID to ensure it belongs to the institute. Users without a valid institute domain Gmail ID shall receive a specific error message indicating the reason for authentication failure. (all)
2. The platform should enable the club coordinators to schedule events for any given date and time. While applying for the same, they should be permitted to edit the event details and registration requirements. (Club Coordinators)
3. The platform would also have dedicated pages for the clubs, which the respective club's coordinator would manage. A club coordinator should be allowed to edit this page by editing club information, images from past events and point of contact information. (Club coordinators)
4. The admin should be given access to a page that lists all pending requests for scheduling the events while marking potential conflicts and should be able to approve or decline each one of them. (admin)
5. After the event's approval, it should be added to the platform for registration by students of the institute, and an email notification should be sent to all the students. (Optional) (all)
6. Students should be allowed to search for upcoming events based on their interests and view them. (all)
7. The platform's admin access should be given to an admin who can edit event details, Club Coordinator and respective clubs' and students' accounts information. (Admin)
8. There should be a report feature in the platform to enable everyone to report incidents and issues with the platform. (all)
9. Students should be able to request for the formation of a new club. (all).
10. The approval for new clubs will be done by the admin. (Admin)

Glossary of Terms:

1. Students: Users who study at the Institute
2. Club coordinators: Students who are responsible for managing the club activities and hosting events.
3. Events coordinator: They are student users responsible for managing and approving the events added by club coordinators/heads.
4. Admin: Appointed by the institute and is responsible for approving events, ensuring their management, etc. can change logins and credentials, and manage the application.
5. Credentials: information required for logging in; in this case, it is the Google account associated with the institute.

