

# **ClubHouse**

## **SYSTEM DESIGN DOCUMENT**

**Tharuth Attanayake**

**Noah Cristino**

**Priyank Dave**

**Amy Li**

**Faraz Kaleem Malik**

**Arailym Mussilim**

**Dhruv Patel**

# Contents

## Front-End

AuthCheck	4
AllClubs (Student View)	4
AllClubsCard (Student View)	4
AnnouncementCard	5
ClubAdminEditProfile	5
ClubAdminMain	5
ClubAdminProfilePage	5
ClubAdminViewProfile	6
ClubApplyButton	6
ClubCard	6
ClubRegisterReq (Form)	6
ClubRequestCard	7
Clubs	7
EmailSetting	7
EmailSettingsPage	7
EventCard (Club View)	8
EventForm (Club View)	8
EventTag	8
Events	9
Home	9
Login	9
LoginPage	10
MyAccounts	10
MyClubs	10
MyClubsCard	10
MyEvents	11
NavBar (Navigation Bar)	11
NewAnnouncement	11
NewClubReqConfirmation	11
NotificationMenu	12
PositionCard	12
PositionCreate	12
PositionForm	12
Positions	13
PotentialMemCard	13
RegisterForm	13
SCSUClubCard	14

SCSUClubs	14
SCSUConfirmation	14
SCSUclubreview	14
StudentClubProfile	15
StudentEventCard	15
Tag	15
TagsInput	15
TestLogin	15

## **Back-End**

Announcement	16
Club	16
Club Applicant	16
Club Member	17
Club Registration Request	17
Club Tag	17
Event	17
Position	18
Potential User	18
Tag	18
User	18
adminProfileDAO	19
clubDAO	19
cookieDAO	19
emailWrapper	19
membershipDAO	20
userDAO	20

## **Software Architecture**

Three-Tier Architecture	21
System Decomposition	21

# Front-end

<b>Class Name:</b> AuthCheck	
<b>Parent Class (if any):</b> React.Component <b>Subclasses (if any):</b>	
<b>Responsibilities:</b> <ul style="list-style-type: none"><li>• Checks if user is allowed to access a page</li><li>• Redirects to a given url if not allowed</li></ul>	<b>Collaborators:</b>

<b>React Component(s):</b> AllClubs (Student View)	
<b>Parent Class (if any):</b> React.Component <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"><li>• Provides megaview for club</li><li>• Check database for clubs</li><li>• Display club information from above</li><li>• Consists of:<ul style="list-style-type: none"><li>○ Club Banner</li><li>○ Club Name</li><li>○ Club Phone Number</li><li>○ Club Email</li><li>○ And a View Club button</li></ul></li></ul>	<b>Collaborators:</b> <ul style="list-style-type: none"><li>• ClubsApplyButton</li><li>• AuthCheck</li></ul>

<b>React Component(s):</b> AllClubsCard (Student View)	
<b>Parent Class (if any):</b> React.Component <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"><li>• Provides format for AllClubsPage</li><li>• Consists of:<ul style="list-style-type: none"><li>○ Club Banner</li><li>○ Club Name</li><li>○ Club Phone Number</li><li>○ Club Email</li><li>○ And a View Club button</li></ul></li></ul>	<b>Collaborators:</b>

<b>React Component(s):</b> AnnouncementCard	
<b>Parent Class (if any):</b> React.Component <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>Provides format for NewAnnouncement</li> </ul>	<b>Collaborators:</b>

<b>React Component(s):</b> ClubAdminEditProfile	
<b>Parent Class (if any):</b> N/A <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>Allow club admin to edit the profile page including, contact info, description of club, and profile picture.</li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>AuthCheck</li> </ul>

<b>React Component(s):</b> ClubAdminMain	
<b>Parent Class (if any):</b> N/A <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>Allow the Club Admin to view and scroll existing members of the club</li> <li>Allow the Club Admin to accept and deny users that have applied to join the club</li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>AuthCheck</li> </ul>

<b>React Component(s):</b> ClubAdminProfilePage	
<b>Parent Class (if any):</b> N/A <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>Allow the Club Admin to go into edit mode or remain in view profile mode.</li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>ClubAdminViewProfile</li> <li>ClubAdminEditProfile</li> <li>AuthCheck</li> </ul>

<b>React Component(s):</b> ClubAdminViewProfile	
<b>Parent Class (if any):</b> N/A <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>Displays the current profile page of the club associated with the club admin</li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>AuthCheck</li> </ul>

<b>React Component(s):</b> ClubApplyButton	
<b>Parent Class (if any):</b> React.Component <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>Access the applyMember endpoint onclick</li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>AllClubs</li> <li>AuthCheck</li> </ul>

<b>React Component(s):</b> ClubCard	
<b>Parent Class (if any):</b> React.Component <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>Provides format for Clubs</li> <li>Includes styling</li> </ul>	<b>Collaborators:</b>

<b>React Component(s):</b> ClubRegisterReq (Form)	
<b>Parent Class (if any):</b> N/A <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>Knows the club name</li> <li>Knows the club email</li> <li>Knows the club phone number (if entered)</li> <li>Knows the club tags</li> <li>Knows the club description</li> <li>If the submit button is clicked, the entered data is validated, then redirected to a confirmation page</li> <li>If input is invalid, an error message is displayed</li> <li>If the cancel button is clicked, the user is redirected to the home page</li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>AuthCheck</li> </ul>

<b>React Component(s):</b> ClubRequestCard	
<b>Parent Class (if any):</b> React.Component <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Provides format for ClubRegisterReq</li> <li>• Includes styling</li> </ul>	<b>Collaborators:</b>

<b>React Component(s):</b> Clubs	
<b>Parent Class (if any):</b> React.Component <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Displays a grid of all clubs</li> <li>• Contains: <ul style="list-style-type: none"> <li>○ Club Name</li> <li>○ Club Description</li> <li>○ Phone</li> <li>○ Email</li> <li>○ Tags</li> <li>○ Icon/Image</li> </ul> </li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>• AuthCheck</li> <li>• ClubCard</li> </ul>

<b>React Component(s):</b> EmailSetting	
<b>Parent Class (if any):</b> React.Component <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Provides component for EmailSettingsPage</li> </ul>	<b>Collaborators:</b>

<b>React Component(s):</b> EmailSettingsPage	
<b>Parent Class (if any):</b> React.Component <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Provides settings page for email notification for users from clubs</li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>• AuthCheck</li> <li>• EmailSetting</li> </ul>

<b>React Component(s):</b> EventCard (Club View)	
<b>Parent Class (if any):</b> N/A <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Displays sections for eventName, organizer's name, start and end times, location, tags, more info in collapse card component</li> <li>• Reformats date, start and end times to display in "mmmm dd, yyyy" and "HH:mm" (shortTime), respectively</li> </ul>	<b>Collaborators:</b>

<b>React Component(s):</b> EventForm (Club View)	
<b>Parent Class (if any):</b> N/A <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Knows the logged in club's name</li> <li>• Knows the event name input</li> <li>• Knows the event image input</li> <li>• Knows the event start time input</li> <li>• Knows the event end time input</li> <li>• Knows the event location input</li> <li>• Knows the event description input</li> <li>• Knows the event's attendees</li> <li>• Knows the event tags input</li> <li>• When submit button is clicked, a post request gets submitted using api endpoint in events router, and then redirect to club profile</li> <li>• When delete button is clicked, redirect to previous page</li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>• AuthCheck</li> </ul>

<b>React Component(s):</b> EventTag	
<b>Parent Class (if any):</b> React.Component <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Provides format for EventTags</li> <li>• Provides styling for EventTags</li> </ul>	<b>Collaborators:</b>



<b>React Component(s):</b> Events	
<b>Parent Class (if any):</b> React.Component <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Provides mega-gridview for all events</li> <li>• Accessible to general users</li> <li>• Contains: <ul style="list-style-type: none"> <li>○ Sorting Options</li> <li>○ Club Name</li> <li>○ Event Name</li> <li>○ Event Date and Time</li> <li>○ Related Tags</li> <li>○ List of Attendees</li> <li>○ Option to join</li> </ul> </li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>• AuthCheck</li> <li>• EventCard</li> </ul>

<b>React Component(s):</b> Home	
<b>Parent Class (if any):</b> React.Component <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Home Page</li> </ul>	<b>Collaborators:</b>

<b>React Component(s):</b> Login	
<b>Parent Class (if any):</b> React.Component <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Allows the user to input an email and password and login</li> <li>• Logs user in and sets cookie if credentials match database</li> <li>• Knows the current state of the register form</li> <li>• Redirects user to home page if already logged in</li> <li>• Displays error message if password is incorrect or if the user doesn't exist</li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>• AuthCheck</li> </ul>

<b>Class Name:</b> LoginPage	
<b>Parent Class (if any):</b> <b>Subclasses (if any):</b>	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Provide Login Container</li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>• AuthCheck</li> <li>• Login</li> </ul>

<b>Class Name:</b> MyAccounts	
<b>Parent Class (if any):</b> <b>Subclasses (if any):</b>	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Showing user events</li> <li>• Showing user clubs</li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>• MyEvents</li> <li>• MyClubs</li> </ul>

<b>React Component(s):</b> MyClubs	
<b>Parent Class (if any):</b> React.Component <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Provides minimal view of all clubs that user is part of</li> <li>• Check database for registered club members</li> <li>• Display posting information from above consisting of the club name and a hyperlink to the club</li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>• AuthCheck</li> </ul>

<b>React Component(s):</b> MyClubsCard	
<b>Parent Class (if any):</b> React.Component <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Format for the Clubmap in "MyClubs" page</li> <li>• Includes minimal styling</li> </ul>	<b>Collaborators:</b>

<b>Class Name:</b> MyEvents	
<b>Parent Class (if any):</b> <b>Subclasses (if any):</b>	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• showing user events</li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>• EventCard</li> <li>• AuthCheck</li> </ul>

<b>React Component(s):</b> NavBar (Navigation Bar)	
<b>Parent Class (if any):</b> React.Component <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Sets navbar components</li> <li>• Changes color when clicked on different components</li> </ul>	<b>Collaborators:</b>

<b>React Component(s):</b> NewAnnouncement	
<b>Parent Class (if any):</b> N/A <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Knows the announcement subject</li> <li>• Knows the announcement message</li> <li>• Knows the recipients</li> <li>• When send button is pressed, a post request is made using api endpoint in announcements route, and then user is redirected to home page</li> <li>• When cancel button is pressed, clear form and redirect to home page</li> </ul>	<b>Collaborators:</b>

<b>React Component(s):</b> NewClubReqConfirmation	
<b>Parent Class (if any):</b> N/A <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Provides a confirmation that the club registration request has been sent</li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>• AuthCheck</li> </ul>

<b>React Component(s):</b> NotificationMenu	
<b>Parent Class (if any):</b> N/A <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Notification Menu</li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>• AuthCheck</li> </ul>

<b>React Component(s):</b> PositionCard	
<b>Parent Class (if any):</b> React.Component <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Format for the Position map in “Positions” page</li> <li>• Includes minimal pseudo-styling</li> </ul>	<b>Collaborators:</b>

<b>React Component(s):</b> PositionCreate	
<b>Parent Class (if any):</b> React.Component <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Provides tab to create a new job listing</li> <li>• Inserts new position into database</li> <li>• Consists of: <ul style="list-style-type: none"> <li>○ Job Position</li> <li>○ Club Name</li> <li>○ Club Email</li> <li>○ Job Description</li> <li>○ Job Requirements</li> <li>○ Club Image uploader</li> </ul> </li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>• AuthCheck</li> <li>• PositionCard</li> </ul>

<b>React Component(s):</b> PositionForm	
<b>Parent Class (if any):</b> React.Component <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Format for the Position form in “PositionCreate” page</li> <li>• Includes styling</li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>• AuthCheck</li> </ul>

<b>React Component(s):</b> Positions	
<b>Parent Class (if any):</b> React.Component <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Provides megaview for all job postings</li> <li>• Check database for positions</li> <li>• Display posting information from above</li> <li>• Consists of: <ul style="list-style-type: none"> <li>○ Job Position</li> <li>○ Club Name</li> <li>○ Club Email</li> <li>○ And a View Club button</li> </ul> </li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>• AuthCheck</li> <li>• PositionForm</li> </ul>

<b>React Component(s):</b> PotentialMemCard	
<b>Parent Class (if any):</b> React.Component <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Provides format and styling for potential members for ClubAdminMain</li> <li>• Uses DB</li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>• Login</li> <li>• AuthCheck</li> </ul>

<b>React Component(s):</b> RegisterForm	
<b>Parent Class (if any):</b> React.Component <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Lets user input username and password</li> <li>• Shows verification input field when required</li> <li>• Allows user to input verification code</li> <li>• Once verified, redirects user to the login screen</li> <li>• Perform basic verification on registration details</li> <li>• If any error, show them on corresponding fields at tiny subtext</li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>• Login</li> <li>• AuthCheck</li> </ul>

<b>Class Name:</b> SCSUClubCard	
<b>Parent Class (if any):</b> <b>Subclasses (if any):</b>	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Format for the Position map in “SCSUClubs” page</li> <li>• Includes minimal pseudo-styling</li> </ul>	<b>Collaborators:</b>

<b>React Component(s):</b> SCSUClubs	
<b>Parent Class (if any):</b> N/A <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Formatted like “Clubs” page</li> <li>• Provides/Requires admin access to delete a club entirely in every collection</li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>• AuthCheck</li> <li>• SCSUClubCard</li> </ul>

<b>Class Name:</b> SCSUConfirmation	
<b>Parent Class (if any):</b> <b>Subclasses (if any):</b>	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Provides confirmation for club registration requests from SCSUclubreview page</li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>• AuthCheck</li> <li>• SCSUclubreview</li> </ul>

<b>Class Name:</b> SCSUclubreview	
<b>Parent Class (if any):</b> <b>Subclasses (if any):</b>	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Showing club registration requests</li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>• AuthCheck</li> <li>• SCSUConfirmation</li> </ul>

<b>React Component(s):</b> StudentClubProfile	
<b>Parent Class (if any):</b> N/A <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>Used for student club profile.</li> </ul>	<b>Collaborators:</b>

<b>React Component(s):</b> StudentEventCard	
<b>Parent Class (if any):</b> N/A <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>Displays eventName, organizer's name, start and end times, location, tags, more info in collapse card component</li> <li>Reformats date, start and end times to display in "mmmm dd, yyyy" and "HH:mm" (shortTime), respectively</li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>AuthCheck</li> </ul>

<b>React Component(s):</b> Tags	
<b>Parent Class (if any):</b> N/A <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>Component for displaying a single tag</li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>Tags</li> </ul>

<b>React Component(s):</b> TagsInput	
<b>Parent Class (if any):</b> N/A <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>Format and styling for tags</li> </ul>	<b>Collaborators:</b> <ul style="list-style-type: none"> <li>TagsInput</li> </ul>

<b>React Component(s):</b> TestLogin	
<b>Parent Class (if any):</b> N/A <b>Subclasses (if any):</b> N/A	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>Test Login page from Sprint 1</li> </ul>	<b>Collaborators:</b>

# Back-end

<b>Class Name:</b> Announcement	
<b>Parent Class (if any):</b> <b>Subclasses (if any):</b>	
<b>Responsibilities:</b> <ul style="list-style-type: none"><li>• Knows its name</li><li>• Knows its subject</li><li>• Knows its message</li><li>• Knows its recipients</li><li>• Knows its date and time posted</li></ul>	<b>Collaborators:</b>

<b>Class Name:</b> Club	
<b>Parent Class (if any):</b> <b>Subclasses (if any):</b>	
<b>Responsibilities:</b> <ul style="list-style-type: none"><li>• Knows its name</li><li>• Knows its email</li><li>• Knows its categories/tags</li><li>• Knows its club image</li><li>• Knows its phone number</li></ul>	<b>Collaborators:</b>

<b>Class Name:</b> Club Applicant	
<b>Parent Class (if any):</b> <b>Subclasses (if any):</b>	
<b>Responsibilities:</b> <ul style="list-style-type: none"><li>• Knows its name</li><li>• Knows its club name</li><li>• Knows its email</li><li>• Knows its club email</li></ul>	<b>Collaborators:</b>



<b>Class Name:</b> Club Member	
<b>Parent Class (if any):</b> <b>Subclasses (if any):</b>	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Knows its name</li> <li>• Knows its club name</li> <li>• Knows its email</li> <li>• Knows its club email</li> <li>• Knows its notification settings</li> </ul>	<b>Collaborators:</b>

<b>Class Name:</b> Club Registration Request	
<b>Parent Class (if any):</b> <b>Subclasses (if any):</b>	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Knows its name</li> <li>• Knows its password</li> <li>• Knows its email</li> <li>• Knows its categories/tags</li> <li>• Knows its description</li> <li>• Knows its phone number</li> </ul>	<b>Collaborators:</b>

<b>Class Name:</b> Club Tag	
<b>Parent Class (if any):</b> <b>Subclasses (if any):</b>	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Keeps track of club tags</li> </ul>	<b>Collaborators:</b>

<b>Class Name:</b> Event	
<b>Parent Class (if any):</b> <b>Subclasses (if any):</b>	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Knows its name, image and organizer's name</li> <li>• Knows its location, date and time</li> <li>• Knows its description</li> <li>• Knows its attendees</li> <li>• Knows its tags/categories</li> </ul>	<b>Collaborators:</b>

<b>Class Name:</b> Position	
<b>Parent Class (if any):</b> <b>Subclasses (if any):</b>	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Knows its name</li> <li>• Knows its club image</li> <li>• Knows its position</li> <li>• Knows its description</li> <li>• Knows its requirements</li> <li>• Knows its club image</li> <li>• Knows its email</li> </ul>	<b>Collaborators:</b>

<b>Class Name:</b> Potential User	
<b>Parent Class (if any):</b> <b>Subclasses (if any):</b>	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Knows its name</li> <li>• Knows its club email</li> <li>• Knows its email</li> <li>• Knows its club email</li> </ul>	<b>Collaborators:</b>

<b>Class Name:</b> Tag	
<b>Parent Class (if any):</b> <b>Subclasses (if any):</b>	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Keeps track of event tags</li> </ul>	<b>Collaborators:</b>

<b>Class Name:</b> User	
<b>Parent Class (if any):</b> <b>Subclasses (if any):</b>	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Knows its email and password</li> <li>• Know what type of account credentials are attached to</li> </ul>	<b>Collaborators:</b>

<b>Class Name:</b> adminProfileDAO	
<b>Parent Class (if any):</b> <b>Subclasses (if any):</b>	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Performing database operations on clubs and club specific events</li> <li>• Inserting image strings</li> <li>• Querying club profile information</li> <li>• Fetching event details specific to the club</li> </ul>	<b>Collaborators:</b>

<b>Class Name:</b> clubDAO	
<b>Parent Class (if any):</b> <b>Subclasses (if any):</b>	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Performing database operations on club members and club applicants</li> <li>• Fetching club members and club applicants</li> <li>• Removing club applicants on deny</li> <li>• Inserting club members on accept</li> </ul>	<b>Collaborators:</b>

<b>Class Name:</b> cookieDAO	
<b>Parent Class (if any):</b> <b>Subclasses (if any):</b>	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Getting and setting cookies</li> </ul>	<b>Collaborators:</b>

<b>Class Name:</b> emailWrapper	
<b>Parent Class (if any):</b> <b>Subclasses (if any):</b>	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>• Sending a verification email</li> </ul>	<b>Collaborators:</b>

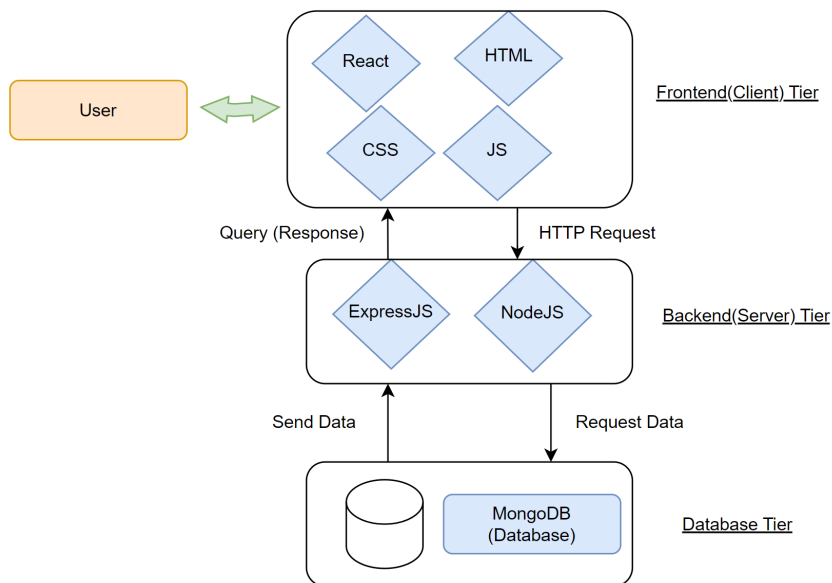
<b>Class Name:</b> membershipDAO	
<b>Parent Class (if any):</b> <b>Subclasses (if any):</b>	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>Finding and adding club membership applications in the database</li> </ul>	<b>Collaborators:</b>

<b>Class Name:</b> usersDAO	
<b>Parent Class (if any):</b> <b>Subclasses (if any):</b>	
<b>Responsibilities:</b> <ul style="list-style-type: none"> <li>Performing database operations on users and potentialUsers</li> <li>inserting</li> <li>finding</li> <li>deleting</li> </ul>	<b>Collaborators:</b>

# Software Architecture

## Three-Tier Architecture

This project utilized the Three-Tier Architecture (MERN Stack). The user interacts with React (the client) frontend of the web application. Any user requests will enable React to send a HTTP request to the application server which is written using NodeJS and ExpressJS. The server then serves the client by accessing the MongoDB database.



## System Decomposition

The system architecture has three separate parts: the database, the frontend, and the backend. The frontend is what the users will be interacting with. The frontend is developed using React and its components. We also used Material UI for universal styling and creating smoother working components. We also call the API endpoints that the backend provides using the fetch call. This way we can allow for CRUD properties to be introduced in our application. The backend of the app was built using Node.js and Express.js. Express is a popular node framework and has a lot of benefits. One of the ones we used is for writing handlers for http requests at different URL paths (routes). This is how we set up the API endpoints that interact with our database (MongoDB). In our database, we have set up different collections for different objects such as clubs, events, and users. We have implemented input sanitation code to make sure we don't receive invalid inputs. And for invalid images, we have implemented pop-up alerts that show up on the user's screen.