



CS 319 Term Project

CluBil: Student Club Manager

Project Analysis Report

İdil YILMAZ - 21703556

Doğuakan Ertunga KURNAZ - 21702331

Turgut Alp EDIS - 21702587

Taha Batur ŞENLİ - 21901857

Argun MURADOV - 21901305

Cemhan Kaan ÖZALTAN - 21902695

Instructor: Eray Tüzün

Teaching Assistant(s): Elgun Jabrayilzade, Emre Sülün, Erdem Tuna, Muhammad Umair Ahmed and Cevat Aykan Sevinç

Oct 31, 2021

This report is submitted to the Department of Computer Engineering of Bilkent University in partial fulfillment of the requirements of the Object-Oriented Software Engineering course CS319.

Contents

1. Introduction	3
2. Proposed System	3
2.1 Overview	3
2.2 Functional Requirements	4
2.2.1 Landing Page	4
2.2.2 Login Page	5
2.2.3 Events List Page	5
2.2.4 Calendar Of Registered Events	6
2.2.5 Student Profile Page	6
2.2.6 Club's Landing Page	6
2.2.7 View Clubs Page	6
2.2.8 Club Profile Page	7
2.2.9 Sysadmin Account	8
2.3 Non-functional Requirements	8
2.4 Pseudo Requirements	9
3. System Models	10
3.1 Scenarios	10
3.2 Use-Case Diagram	21
3.3 Object and Class Diagram	22
3.4 Activity Diagrams	23
3.5 Sequence Diagrams	25
3.6 State Diagrams	28
3.7 Mockup Diagrams (UI Models)	30
Glossary	48
References	49

1 Introduction

The program we've developed is a web-based application designed to help Bilkent University students and student clubs. Our team created this project with hopes of easing the interaction between student clubs, students and the Student Activities Center (SAC). As such, the program can be used by students, board members of clubs and administrators of SAC to hasten the process of organizing, approving and announcing club events.

The application provides students a neat and quick access to all the club events planned in the university as well the chance to enroll in clubs or certain activities. On the other hand, student clubs, alongside managing their clubs' profiles, can quickly create events and submit application forms to SAC. Finally, SAC can use the system to easily coordinate events as well as student clubs.

2 Proposed System

This is where you provide the details of the results of your analysis work.

2.1 Overview

With the program, student clubs can manage their members, activities, profiles and even budgets. Students, on the other hand, can join clubs, enroll in activities and customize their event calendars as well as their profiles. To put into perspective, the following bullet point list refers to all the basic features of the program:

Using the program, Bilkent University students can:

- o Customize their profiles.
- o Join or leave student clubs.
- o Register to or unregister from student club activities.
- o Receive private invitations to club-restricted activities from student clubs they are a part of.
- o Personalize a monthly calendar which outlines all the activities they have registered to.

- o List activities open to everyone within Bilkent University and the activities specific to the student clubs they are a part of.

Using the program, Bilkent University student clubs can:

- o Customize their student club profiles.
- o Create activities that need to be approved by SAC.
- o Delete approved activities or activity submissions waiting to be approved by the SAC.
- o Inform members of their clubs by sending private messages to students.
- o Access a simple budget panel designed to note down the club's spending as well as sponsorships and sources of revenue.

Using the program, Bilkent University SAC administrators can:

- o Add or delete students.
- o Add or delete student clubs.
- o Approve or reject, thus delete, student club activities.
- o Send or receive private messages from student clubs.
- o View a list of all student clubs and all the planned student club activities.
- o View each student club's budget.

· Additionally, the program's main page allows anyone to see:

- o The activities organized by student clubs that are marked as open to everyone, such as large summits.

2.2 Functional Requirements

2.2.1 Landing Page

The first screen the program's users will see will be the landing page, which will display all the upcoming public events open to everyone. It will also

include online events marked as public alongside links to the external meeting app, such as Zoom. We added this feature because it will help clubs with promoting activities open to other universities. Examples to this case are large summits or activities organized alongside fellow clubs from other universities. This page will also include a login button taking the user to the application's login feature.

2.2.2 Login Page

The login page of the program will be accessible from the top bar of the landing page for the user. Here, two login options will be displayed side by side, one of which is for students and the other for clubs. The student login will have text fields for taking a student ID and a password and the club login will be done with an assigned username and password combination. In addition, a sysadmin login which will not be shown explicitly will also be available for the usage of the Student Activities Center and other administrative personnel. All account creations will be done through the sysadmin account, similar to the STARS system of Bilkent University, to prevent unnecessary account flooding.

2.2.3 Events List Page

The Events List Page will be the first page seen by logged in students on the platform. Similar to the GE250/251 page in the STARS system, this page will display all upcoming university-restricted face-to-face and online activities in list form. The user will be able to view information such as the name, date, time, duration, location (platform name for online events), the organizing club and the total/available quota for the event.

The list will be in chronological order, starting from the closest event. Filters at the top of the page will let the student filter the events by keyword and filter online or face-to-face activities specifically.

The “show more” button within the list will show a pop-up window, with additional description of the event, relevant URLs (registration form, Zoom

link etc.), and the “Register to Event” button through which a student can register to the event.

2.2.4 Calendar Of Registered Events

In the Calendar page, students will be able to view their registered events in a monthly calendar view. The events represented in the calendar will be clickable buttons which will again display the “show more” pop-up described in the previous section. Students can use this popup to cancel their registration to the event and delete the event from their calendar.

2.2.5 Student Profile Page

In the profile page, the students will be able to view their personal information, profile picture, joined clubs and relevant tags (if they have a specific position within the club), and an “about me” section where they can add a biography. In this page, students can also see their inbox with messages and club-restricted event invites from their clubs.

2.2.6 Club’s Landing Page

This page performs the same as the 2.2.3 Landing Page written under the Student title on the top side. In Landing Page, in addition to Bilkent University students, students of other universities(such as Hacettepe University, ODTU or Bogazici University) can find activities for themselves. Since these public activities at the Landing Page can be online via Zoom and are open for participation by many people, every student can easily use it. From the point of view of clubs, it will be beneficial to advertise their own clubs,because landing page will be a field that brings together the supply and demand of both university students and clubs.

2.2.7 View Clubs Page

In a separate page, students can look throughout all the active clubs within the university in list form. They can also filter this list to clubs they have joined already, or simply by searching for a club. The “show more” button for each club will open a pop-up that will display the logo, purpose and

upcoming activities for the specific club. The students will also be able to Join or Leave the club throughout this pop-up window.

2.2.8 Club Profile Page

In the profile section of the club, there will be a club logo and/or a photo of the club managers and informative information about the club for students who want to participate in events. The ability to edit a profile and change a password just below the club name will become available to club administrators. When the club manager wants to edit the profile, the Edit Profile button is clicked, and then the profile name, student ID, Club information, and profile photo can be edited in the pop-up that opens. The club manager who wants to make a password change can type his/her new password after confirming his/her old password with the pop-up that appears when s/he clicks the button.

In a frame where the Upcoming events are visible, the club administrator can create a new event for the club(Create Event), as well as delete one or more of the created events(Delete Event) .Each time a new event is created, the club to which the event belongs, the date it will be held and the time between which it will take place can be seen in the Upcoming Events frame, if the user wants to get more detailed information about any event, More section can be used.If the club administrator wants to add an event, clicking on the Add Event button will open a pop-up.The event name, date and time of the event, location and description information; quota for how many people the event will be, type of event (public, campus only or private), photo of the event (such as banner) will be in this pop-up. If the club administrator wants to delete an event, clicking on the Delete Event button opens a pop-up and lists all the events that can be deleted here.

The club's finances can be checked from the "Finance" tab on the top bar. This page, also visible by the SAC admin, contains detailed information about the club's sources of revenue, expenditures and the cost of each event. It also has a pie chart that displays how the club's total budget is appropriated to the events. The club user can add or delete events in this panel. When adding an event, its initial budget and how much of that budget is actually used can be specified as well. On the other hand, the SAC admin has access

to this panel. They can view how the club's budget is spent to decide whether to approve or delete the club's proposed events. Additionally, the SAC administrator can edit a club's total budget, increasing or decreasing it. However, this is done from the SAC admin panel and the SAC admin must provide a reason for the change.

2.2.9 Sysadmin Account

The sysadmin account will mainly be used by the Student Activities Center employees for administrative purposes. When a new student is admitted to the university, their account will be created through this sysadmin account with their student ID as the username and an initial password which may later be changed. Similarly, when a new club is formed, their registration will be made through the sysadmin account. This account will also be able to delete the listed types of accounts in the case of a student leaving the university or a club being closed. Organized student club events will also be confirmed and created by administrative personnel through this account, along with the deletion of denied events.

2.3 Non-functional Requirements

Reliability

- o The system will warn the user if the time of two different registered events collide. However, this warning can be overridden.
- o The system will warn the club user if they try to schedule two different events at the same time. However, this warning can be overridden.
- o The system will display a warning message if the club user tries to use more than the remaining budget while creating an activity in the finances tab.
- o The system will refuse an event form filled by the club if the club enters a location that's already been reserved.
- o The system will deny entry and return an error message if the user's login credentials are incorrect.

- o The system will decline a password change if the user enters their old password.
- o The system will decline a username change if the user enters their old name.
- o The system will provide a warning to the SAC admin if they do not include a reason for the budget change.

Usability

- o The user interface of the program is designed to be highly intuitive and easy to learn.
- o Different kinds of operations are separated into different pages in the User Interface (UI) to make the application easier to understand.
- o General UI design conventions are followed, with the available tabs being similar to other web applications and the visual design aiming to make the usage easier.

Performance

- o The system will automatically log the user out if there is no interaction for more than 6 minutes.
- o Pop ups will open in less than 3 seconds.

Supportability

- o The system will be written to allow additional features through its object oriented design.
- o User issues will be resolved through a help email account.

2.4 Pseudo Requirements

- The system must be programmed using Object-Oriented-Programming (OOP).
- The application must be web-based.
- The system must use GitHub repositories to store the code and track progress.
- The system must be designed using Universal Markup Language (UML).

- The system must satisfy the needs of Bilkent University SAC Administrator.

3. System Models

3.1 Scenarios

1.

- **Use Case :** EditProfile
- **Participating Actor:** Initiated by Student OR
Initiated by Club Manager
- **Entry Condition:**
 - The user logs into the system.
 - The user clicks the name at the top bar.
- **Exit Condition:**
 - The user clicks on "Save Changes" button.
 - The user exits from the window without saving the changes.
- **Flow of events:**
 - The user clicks the "Edit Profile" button.
 - The program responds by opening a popup including the information about the user.
 - The user sets the new information in the screen.
 - The user clicks "Save Changes" button.
 - The program responds by updating the user's information in the profile page and database.
- **Quality Requirements:**
 - The user must not remain inactive for longer than 6 minutes.

2.

- **Use Case :** RequestBudgetChange
- **Participating Actor:** Initiated by Student Club
- **Entry Condition:**
 - Student Club opens the finance tab of the club.
- **Exit Condition:**
 - Student Club submits a budget change request form to the Student Activities Center (SAC).
- **Flow of events:**
 - Student Club clicks on the “Request Budget Change” button on the finance page.
 - The program responds by opening a popup of a form which includes the requested new budget and the reason for the requested change. The old budget is also shown in the popup.
 - The student club fills the form and submits it by clicking on the “Submit” button on the bottom.

- The program responds by sending the request to the SAC Administrator as a notification.
- **Quality Requirements:**
 - The user must not remain inactive for longer than 6 minutes.
 - At the start of the flow of events, this use case includes the OpenManageFinanceTab use case. The OpenManageFinanceTab use case initiates when the user opens their club's finances.

3.

- **Use Case :** JoinEvent
- **Participating Actor:** Initiated by Student
- **Entry Condition:**
 - Students log into the system.
- **Exit Condition:**
 - Student joins the event
- **Flow of events:**
 - Student selects an event from the events list, which is the first page seen after login.
 - The program responds by opening a popup including detailed information about the event.
 - Students click on the “Join” button on the top right corner.
 - The program responds by updating the user’s status in their calendar and also changing the clicked “Join” icon to the “Leave” icon.
- **Quality Requirements:**
 - The user must not remain inactive for longer than 6 minutes.
 - At the start of the flow of events, this use case includes the SeeEventListAsStudent use case. The SeeEventListAsStudent use case initiates when the user opens the list of all events.

4.

- **Use Case :** LeaveEvent
- **Participating Actor:** Initiated by Student
- **Entry Condition:**
 - Student logs into the system OR,
 - Student opens their event calendar.
- **Exit Condition:**
 - Student leaves the event.
- **Flow of events:**
 - Student selects an event.
 - The program responds by opening a popup including detailed information about the event.
 - Student clicks on the “Leave” button on the top right corner.
 - The program responds by updating the user’s status in their calendar and also changing the clicked “Leave” icon to the “Join” icon.
- **Quality Requirements:**

- The user must not remain inactive for longer than 6 minutes.
- At any given point during the flow of events, this use case can include the SeeEventListAsStudent OR SeeEventCalendarAsStudent use cases. The SeeEventListAsStudent use case is initiated when the user opens the list of all events. The SeeEventCalendarAsStudent is initiated when the user clicks on the Calendar tab.

5.

- **Use Case:** JoinClub
- **Participating Actor:** Student
- **Entry Condition:** Student is logged in
 - Student has not joined the selected club before
- **Exit Condition:** Student clicks the join club button
- **Flow of Events:**
 - Student clicks on the “Clubs” button
 - The program displays the club list to the student
 - Student selects the club they would like to join and clicks “Show More” button
 - The program shows the pop-up that shows the detail information about the selected club
 - Student clicks the “join” button in the pop-up
 - The program adds the student to the club and the student joined the club
- **Quality Requirements:**
 - The user must not remain inactive for longer than 6 minutes.
 - It includes the seeClubListasStudent that shows the all clubs to the student.

6.

- **Use Case:** LeaveClub
- **Participating Actor:** Student
- **Entry Condition:** Student is logged in
 - Student is a member of the club
- **Exit Condition:** Student clicks the leave club button
- **Flow of Events:**
 - Student clicks on the “Clubs” button
 - The program displays the club list to the student
 - Student selects the club they would like to leave and clicks “Show More” button
 - The program shows the pop-up that shows the detail information about the selected club
 - Student clicks the “leave” button in the pop-up
 - The program removes the student to the club and the student left the club
- **Quality Requirements:**
 - The user must not remain inactive for longer than 6 minutes.

- It includes the seeClubListasStudent that shows the all clubs to the student.

7.

- **Use Case:** SeeUpcomingClubEvents
- **Participating Actor:** Initiated by Student
- **Entry Condition:**
 - Student opens the list of all clubs tab.
- **Exit Condition:**
 - Student sees the events specific to a single club.
- **Flow of events:**
 - Student selects a club from the list of all clubs tab.
 - The program responds by opening a popup including detailed information about the club and all of the club's upcoming events.
 - From here the student can click on any upcoming event and view more information about them.
- **Quality Requirements:**
 - The user must not remain inactive for longer than 6 minutes.
 - At the start of the flow of events, this use case includes the SeeClubListAsStudent use case. The SeeClubListAsStudent use case initiates when the user opens the list of all clubs registered to the program.

8.

- **Use Case:** AcceptInvitation
- **Participating Actor:** Initiated by Student
- **Entry Condition:** Student is logged in
 - Student has received a private invitation from a club
 - Student is clicked on their name at the top bar
- **Exit Condition:** Student accepts the invitation
- **Flow of Events:**
 - Student clicks on the “My Profile” button
 - The program displays the profile page of the student
 - Student Selects the invitation they would like to respond to from the “Notifications” section
 - Student clicks one of the “Accept” buttons from the view.
 - The program notifies the club account according to the respond from the user.
- **Quality Requirements:**
 - The user must not remain inactive for longer than 6 minutes.
 - The invited event must be not full.

9.

- **Use Case:** DeclineInvitation
- **Participating Actor:** Initiated by Student
- **Entry Condition:** Student is logged in
 - Student has received a private invitation from a club

Student is clicked on their name at the top bar

- **Exit Condition:** Student declines the invitation
- **Flow of Events:**
 - Student clicks on the “My Profile” button
 - The program displays the profile page of the student
 - Student Selects the invitation they would like to respond to from the “Notifications” section
 - Student clicks one of the “Decline” buttons from the view.
 - The program notifies the club account according to the respond from the user.
- **Quality Requirements:**
 - The user must not remain inactive for longer than 6 minutes.

10.

- **Use Case:** SeeEventCalendarAsClub
- **Participating Actor:** Club Manager
- **Entry Condition:** Club manager clicks the Calendar button
- **Exit Condition:** Club Manager goes to another tab.
- **Flow of Events:**
 - Club Manager clicks the Calendar button
 - Club Manager sees the events created by their club on a calendar.

11.

- **Use Case:** ChangePassword
- **Participating Actor:** Student, Club Manager
- **Entry Condition:** Change Password button is clicked on the Profile page
- **Exit Condition:** Password is changed
- **Flow of Events:**
 - Change Password button is clicked
 - User enters their old and new password
 - Password is updated

12.

- **Use Case:** SamePassword
- **Participating Actor:** Student, Club Manager
- **Entry Condition:** This use case extends ChangePassword and is initiated when the user tries setting the new password as the old one
- **Exit Condition:** Error message is displayed.
- **Flow of Events:**
 - User tries setting the new password as the old one
 - Password is not accepted an no updates are made in the database
 - An error message is displayed

13.

- **Use Case:** SeeClubListasClub
- **Participating Actor:** Club Manager
- **Entry Condition:** Club Manager clicks the Clubs button
- **Exit Condition:** Club Manager goes to another page
- **Flow of Events:**
 - Club manager clicks the Clubs button and goes to the Clubs tab
 - All existing clubs are listed

14.

- **Use Case:** SeeEventListAsClub
- **Participating Actor:** Club Manager
- **Entry Condition:** Club Manager clicks the Events button
- **Exit Condition:** Club Manager goes to another page
- **Flow of Events:**
 - Club manager clicks the Events button and goes to the Events tab
 - All existing events are listed

15.

- **Use Case:** AddEventCost
- **Participating Actor:** Club Manager
- **Entry Condition:** Event to add a cost to exists
- **Exit Condition:** Cost is added to the event or the given cost exceeds the budget
- **Flow of Events:**
 - Club Manager clicks the Add Event button on the Finance tab
 - Club Manager enters the event name, cost and the spent amount
 - Event cost is either added or denied due to the cost being higher than total budget

16.

- **Use Case:** MoreCostThanIncome
- **Participating Actor:** Club Manager
- **Entry Condition:** This use case extends AddEventCost and is initiated when the cost of an event exceeds the club budget.
- **Exit Condition:** Event cost is denied
- **Flow of Events:**
 - Club Manager sets the event cost too high during budget addition
 - The event cost is denied and not added to the database
 - An error message is displayed
- **Quality Requirements:**
 - The user must not remain inactive for longer than 6 minutes.

17.

- **Use Case:** RemoveEventCost
- **Participating Actor:** Club Manager
- **Entry Condition:** Event to delete the cost of exists and already has a cost
- **Exit Condition:** Event cost is deleted
- **Flow of Events:**
 - Club Manager clicks the Delete Event button on the Finance tab
 - Club Manager selects the event to delete the cost of
 - Cost of the selected event is deleted
- **Quality Requirements:**
 - The user must not remain inactive for longer than 6 minutes.

18.

- **Use Case:** AddEventRequest
- **Participating Actor:** Club Manager
- **Entry Condition:** Club Manager clicks the Add Event button on the Profile page
- **Exit Condition:** Event proposal is added
- **Flow of Events:**
 - Club Manager clicks the Add Event button on the Profile page
 - Club Manager enters event details on a pop up
 - Event proposal is added for confirmation or deletion by the SAC
- **Quality Requirements:**
 - The user must not remain inactive for longer than 6 minutes.

19.

- **Use Case:** DeleteOwnEvent
- **Participating Actor:** Club Manager
- **Entry Condition:** Club Manager clicks the Delete Event button on the Profile page
- **Exit Condition:** The confirmed event is deleted
- **Flow of Events:**
 - Club Manager clicks the Delete Event button
 - Club Manager selects the event to be deleted
 - Selected event is deleted from the database
- **Quality Requirements:**
 - The user must not remain inactive for longer than 6 minutes.

20.

- **Use Case:** AddClub
- **Participating Actors:** Initiated by SAC Administrator
- **Flow of Events:**
 - SAC Administrator clicks Manage Clubs button.
 - In the Manage Clubs page, there will be clubs with their code and their events' names in addition to their operations.
 - When SAC Administrator clicks Icon with and addition sign
 - The program responds by opening a pop-up which contains profile name, ID, password and User Type for SAC Administrator to fill.

- SAC Administrator fills the form and, in the end, clicks the Add Profile Button.
 - The program receives the form and updates the list of clubs with the new clubs' ID and name.
 - The club's password is also stored afterwards.
- The new profile can be added to the club's list.
- **Entry Condition:**
 - The SAC Administrator opens the Manage Clubs tab.
- **Exit Condition:**
 - The SAC Administrator clicked on another tab.
- **Quality Requirements:**
 - The user must not remain inactive for longer than 6 minutes.

21.

- **Use Case:** DeleteClub
- **Participating Actors:** Initiated by SAC Administrator
- **Flow of Events:**
 - The SAC Administrator opens the Manage Users tab.
 - The program responds by taking the SAC Adminisrtator to the tab.
 - SAC Administrator clicks on the minus icon at the far end of a row for a club.
 - The program responds by removing the club from the database, including all of its events, profile details, finances and password.
- **Entry Condition:**
 - The SAC Administrator opens the Manage Clubs tab.
- **Exit Condition:**
 - The SAC Administrator clicked on another tab.
- **Quality Requirements:**
 - The user must not remain inactive for longer than 6 minutes.

22.

- **Use Case:** SetBudget
- **Participating Actors:** Initiated by SAC Administrator
- **Flow of Events:**
 - The SAC Administrator opens the Manage Clubs tab of the program.
 - The program responds by taking the SAC Administator to the tab.
 - SAC Administrator clicks on the money icon at the end of the row of a club, in operations section.
 - The program responds by opening a popup form for the SAC Administrator to fill.
 - SAC Administrator fills the form.
 - The program receives the form and updates the budget with the new monthly budget with its reason.
 - The club's new budget is also stored afterwards.
- **Entry Condition:**
 - The SAC Administrator opens the Manage Clubs tab.

- **Exit Condition:**
 - The SAC Administrator clicked on another tab.
- **Quality Requirements:**
 - The user must not remain inactive for longer than 6 minutes.

23.

- **Use Case:** ConfirmEvent
- **Participating Actor:** SAC Administrator
- **Entry Condition:** Event is not yet confirmed by the SAC
- **Exit Condition:** Event is confirmed
- **Flow of Events:**
 1. SAC Administrator clicks on the confirm button next to an event
 2. Event is confirmed and reaches the approved status
- **Quality Requirements:**
 - The user must not remain inactive for longer than 6 minutes.
 - The SAC Administrator must click on the Manage Events tab. During the flow of events this invokes OpenManageEventsTab Use Case.

24.

- **Use Case:** DeclineEvent
- **Participating Actor:** SAC Administrator
- **Entry Condition:** Event is not yet confirmed by the SAC
- **Exit Condition:** Event is declined and its proposal is deleted
- **Flow of Events:**
 - SAC Administrator clicks on the decline button next to an event
 - Event is declined and deleted from the database
- **Quality Requirements:**
 - The user must not remain inactive for longer than 6 minutes.

25.

- **Use Case:** DeleteEvent
- **Participating Actor:** SAC Administrator
- **Entry Condition:** Event to be deleted exists.
- **Exit Condition:** Event is deleted.
- **Flow of Events:**
 - SAC Administrator clicks delete event button next to an event.
 - Event is deleted from the database.
- **Quality Requirements:**
 - The user must not remain inactive for longer than 6 minutes.

26.

- **Use Case:** AddUser
- **Participating Actors:** Initiated by SAC Administrator
- **Flow of Events:**
 - The SAC Administrator clicks on the “Add User” button on the top bar, which is marked as the person icon with a plus sign next to it.
 - The program responds by opening a popup form for the SAC Administrator to fill.
 - SAC Administrator fills the form and, in the end, must select the user type as “Student.”
 - The program receives the form and updates the list of students with the new student’s ID and name.
 - The student’s password is also stored afterwards.
- **Entry Condition:**
 - The SAC Administrator opens the Manage Users tab.
- **Exit Condition:**
 - The SAC Administrator clicked on another tab.
- **Quality Requirements:**
 - The user must not remain inactive for longer than 6 minutes.

27.

- **Use Case:** DeleteUser
- **Participating Actors:** Initiated by SAC Administrator
- **Flow of Events:**
 - The SAC Administrator opens the Manage Profiles tab.
 - The program responds by taking the SAC Administator to the tab.
 - SAC Administrator clicks on the minus icon at the far end of a row for a club.
 - The program responds by removing the user from the database, including all of its events, profile details, finances and password.
- **Entry Condition:**
 - The SAC Administrator opens the Manage Users tab.
- **Exit Condition:**
 - The SAC Administrator clicked on another tab.
- **Quality Requirements:**
 - The SAC Administrator must not remain inactive for longer than 6 minutes.

28.

- **Use Case:** SeeClubListAsClub
- **Participating Actor:** Initiated by Club Manager
- **Flow of Events:**
 - Club manager activates the “club list” function by clicking the button in the top bar menu.
 - The program responds by displaying the list of clubs to the club manager.
 - Club manager can choose the club and click the more button at the end column.

- The program responds by displaying the detail information about the selected club.
- **Entry Condition:** The Club Manager clicks the club list button
- **Exit Condition:** The Club Manager clicks another button
- **Quality Requirements:**
 - The response from the club manager has to be received no later than 6 minutes.

29.

- **Use Case:** SeePublicEvents
- **Participating Actor:** Unauthorized User
- **Flow of Events:**
 - Unauthorized User opens the application webpage for the first time in a session
 - The program responds by opening the landing page containing all public events such as Zoom meetings.
- **Entry Condition:** Unauthorized User initially opens the application webpage.
- **Exit Condition:** Unauthorized User sees the events on the landing page.

3.2 Use-Case Diagram

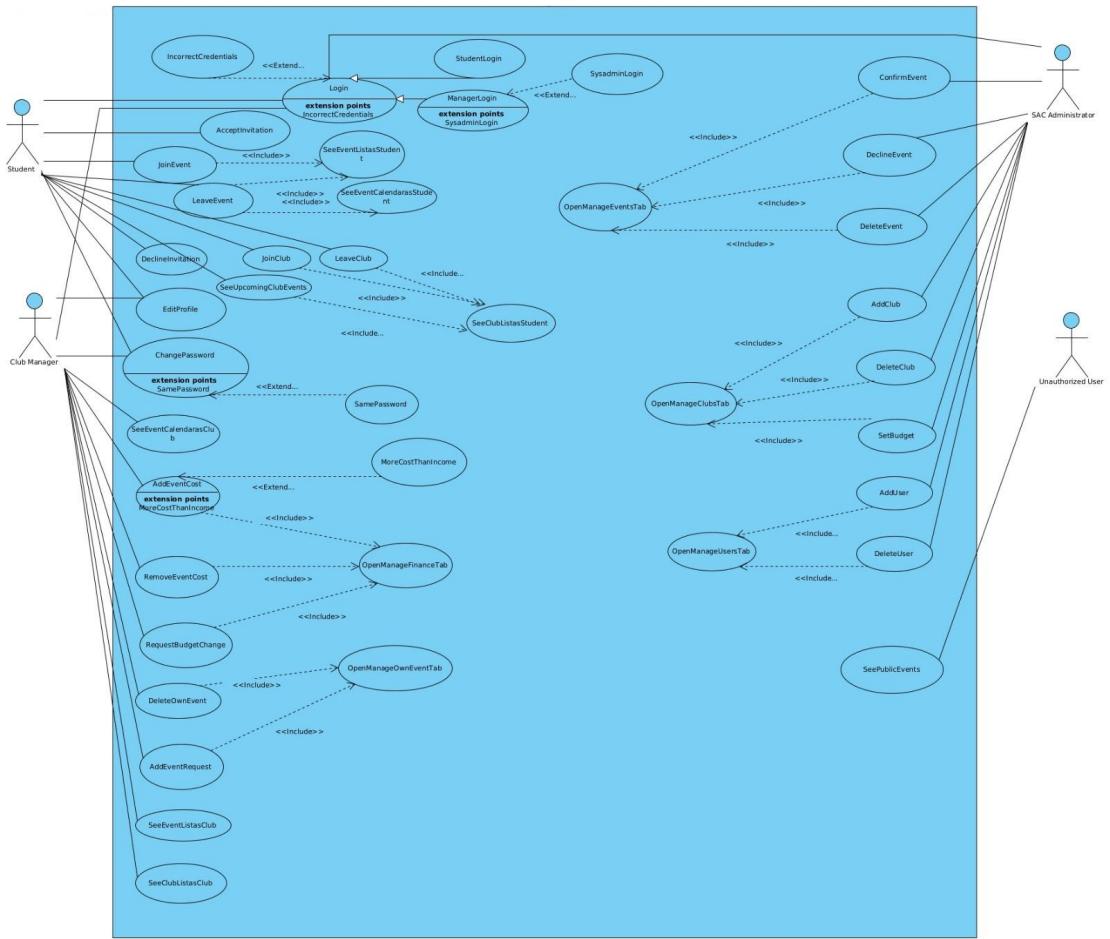


Figure 1: Use Case diagram

3.3 Object and Class Diagram

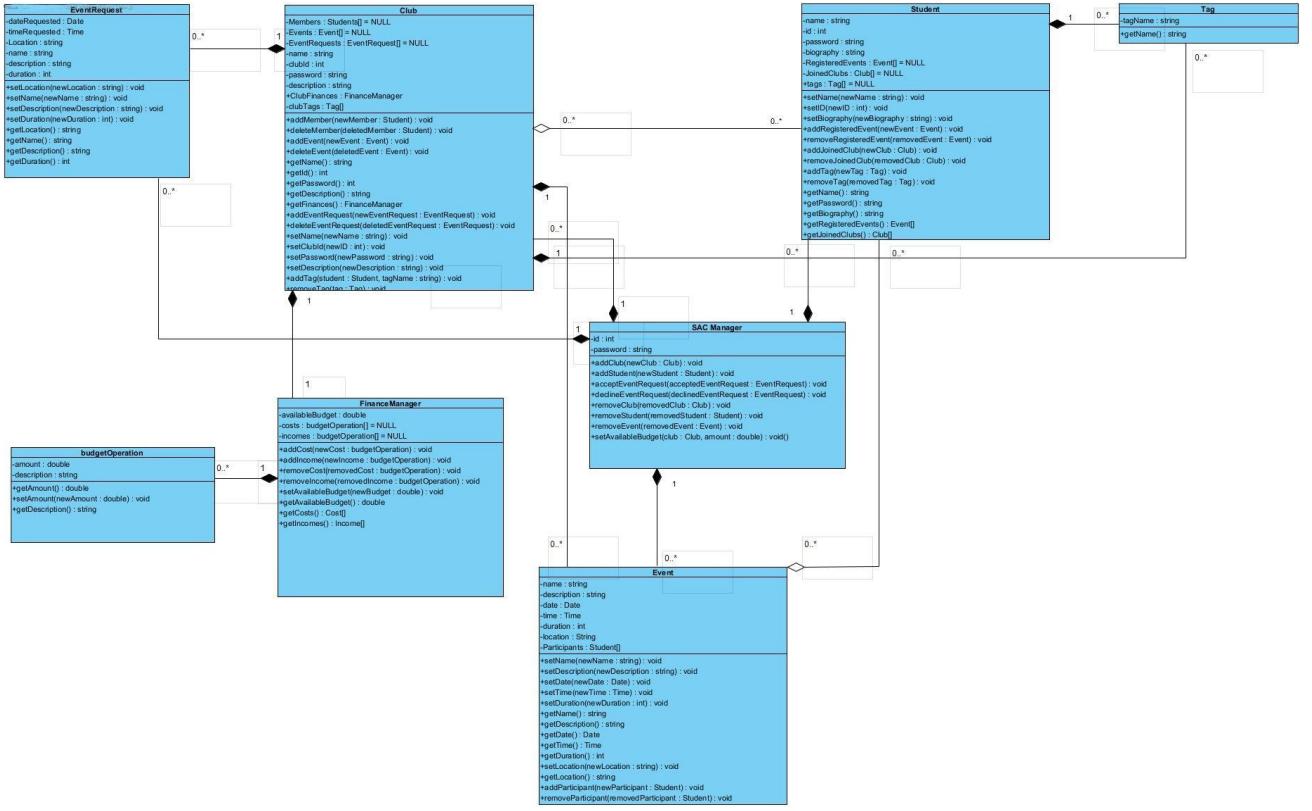


Figure 2: Class diagram

3.4 Activity Diagrams

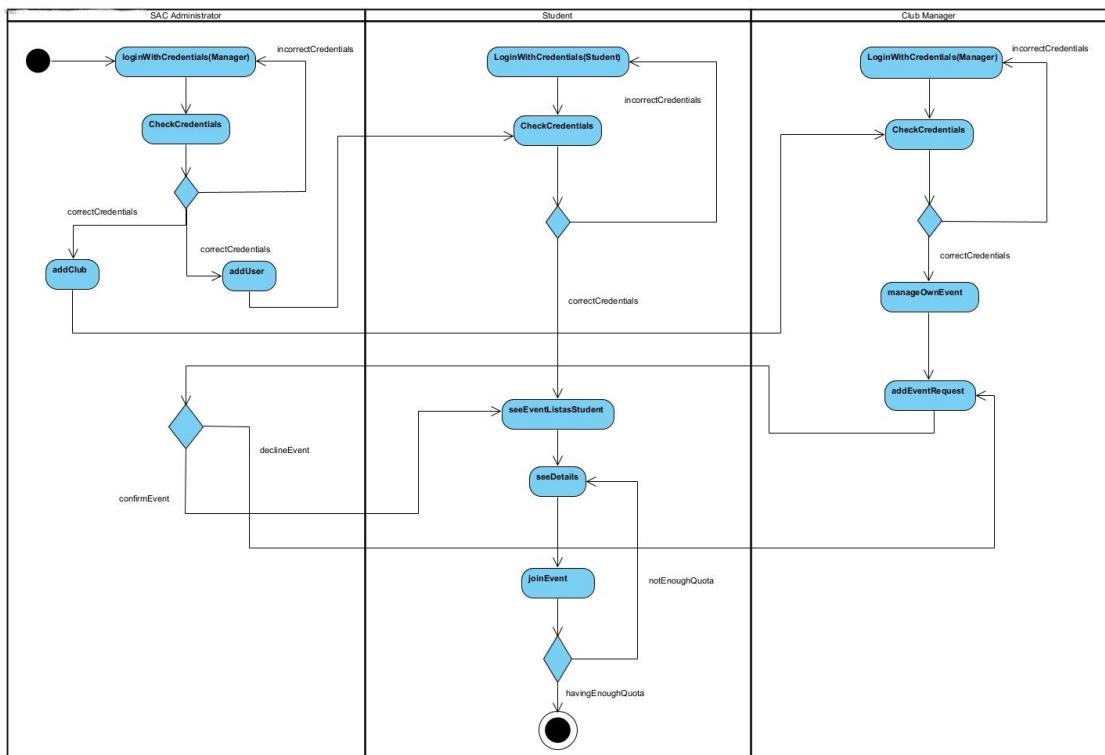


Figure 3: Join Event Activity Diagram

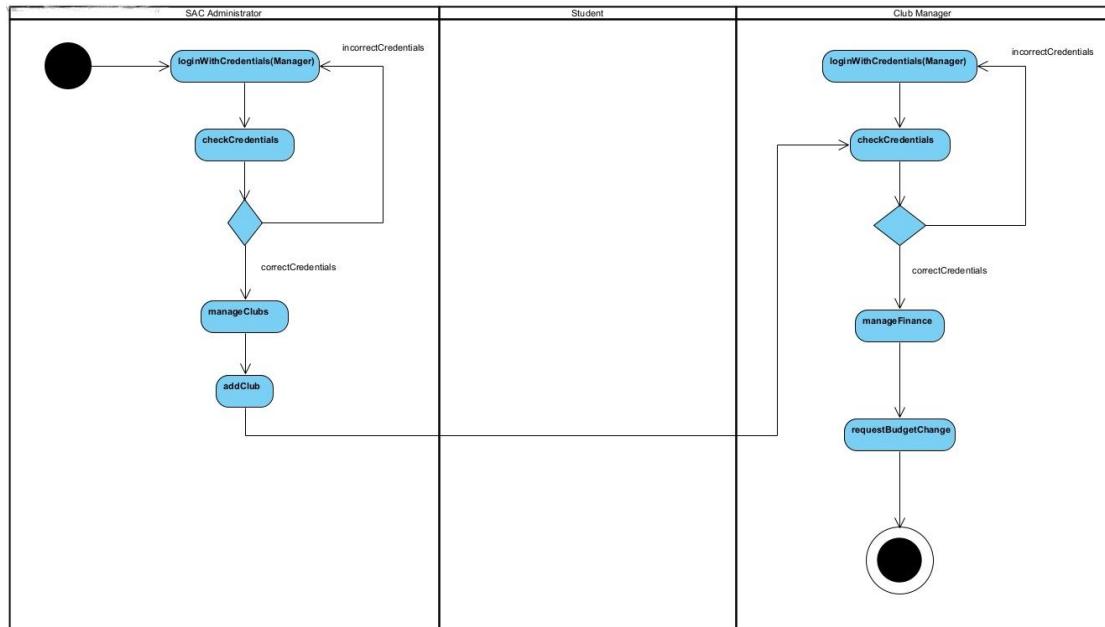


Figure 4: Budget Change Activity Diagram

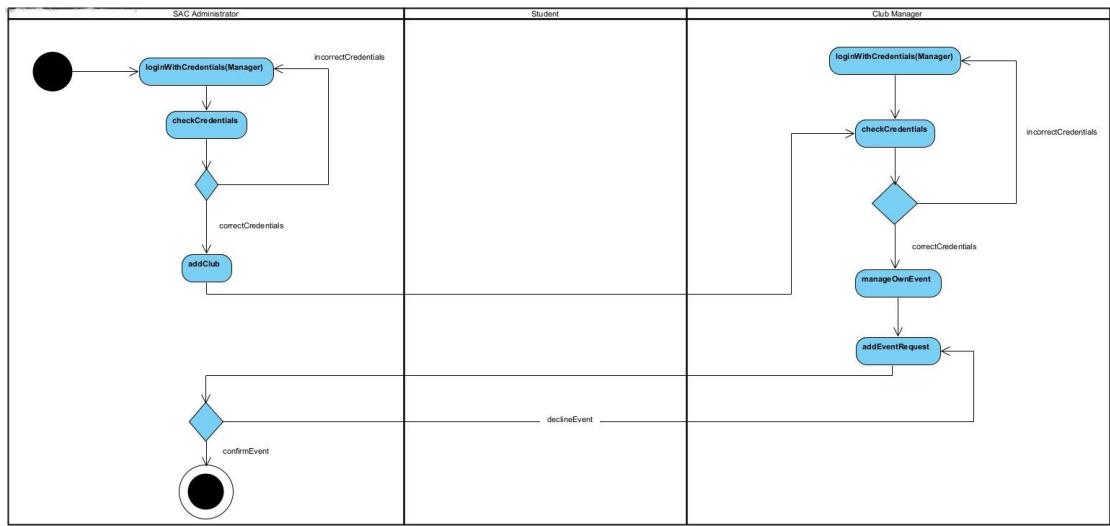


Figure 5: Confirm Event Activity Diagram

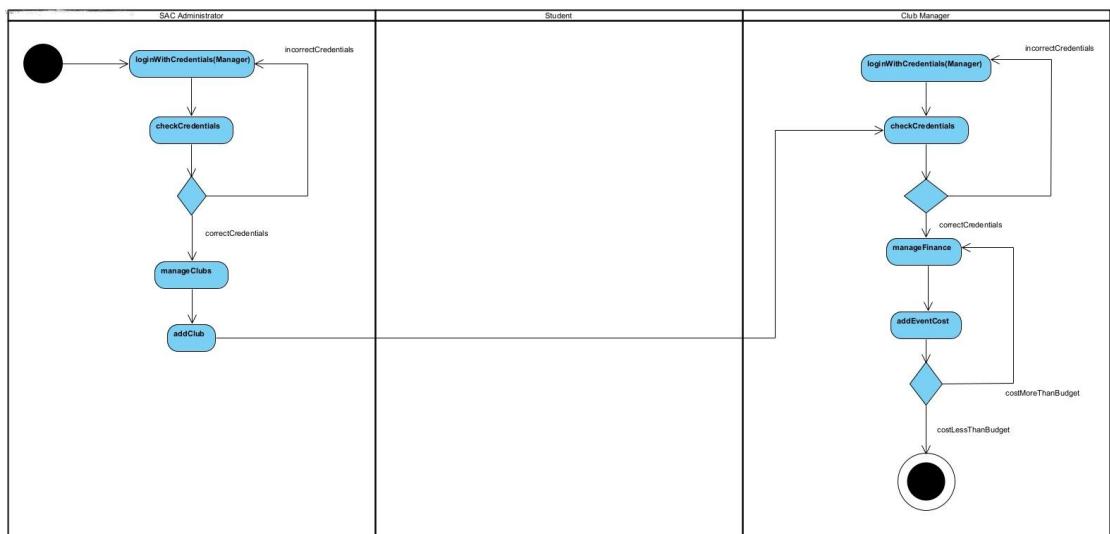


Figure 6: Add Event Cost Activity Diagram

3.5 Sequence Diagrams

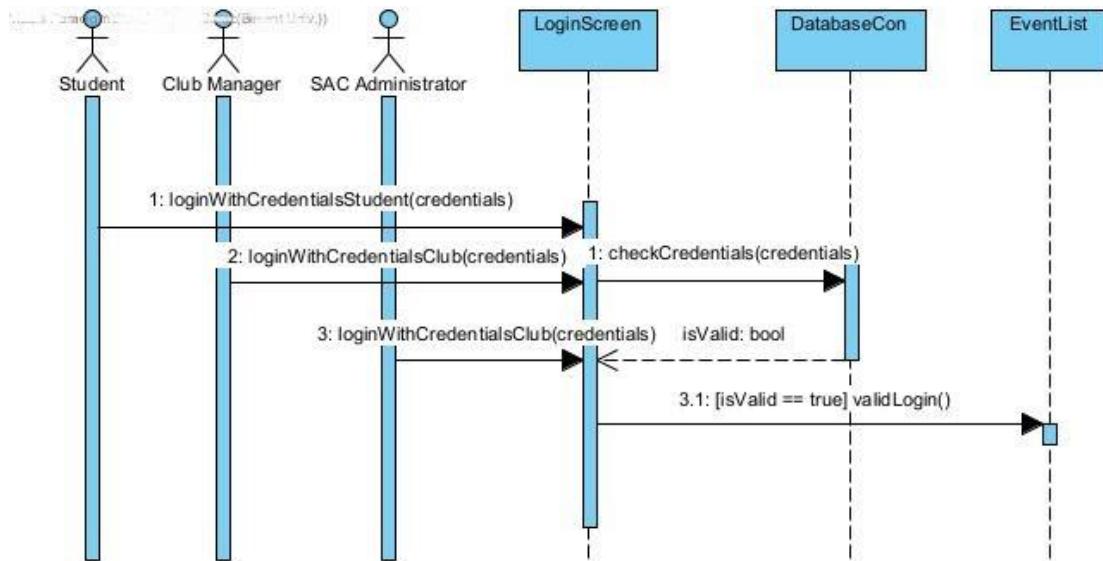


Figure 7: Login Sequence Diagram

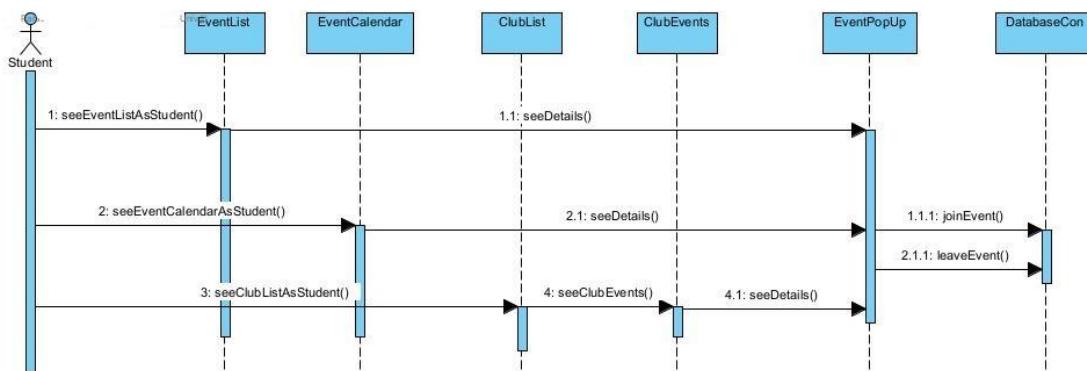


Figure 8: Student Sequence Diagram

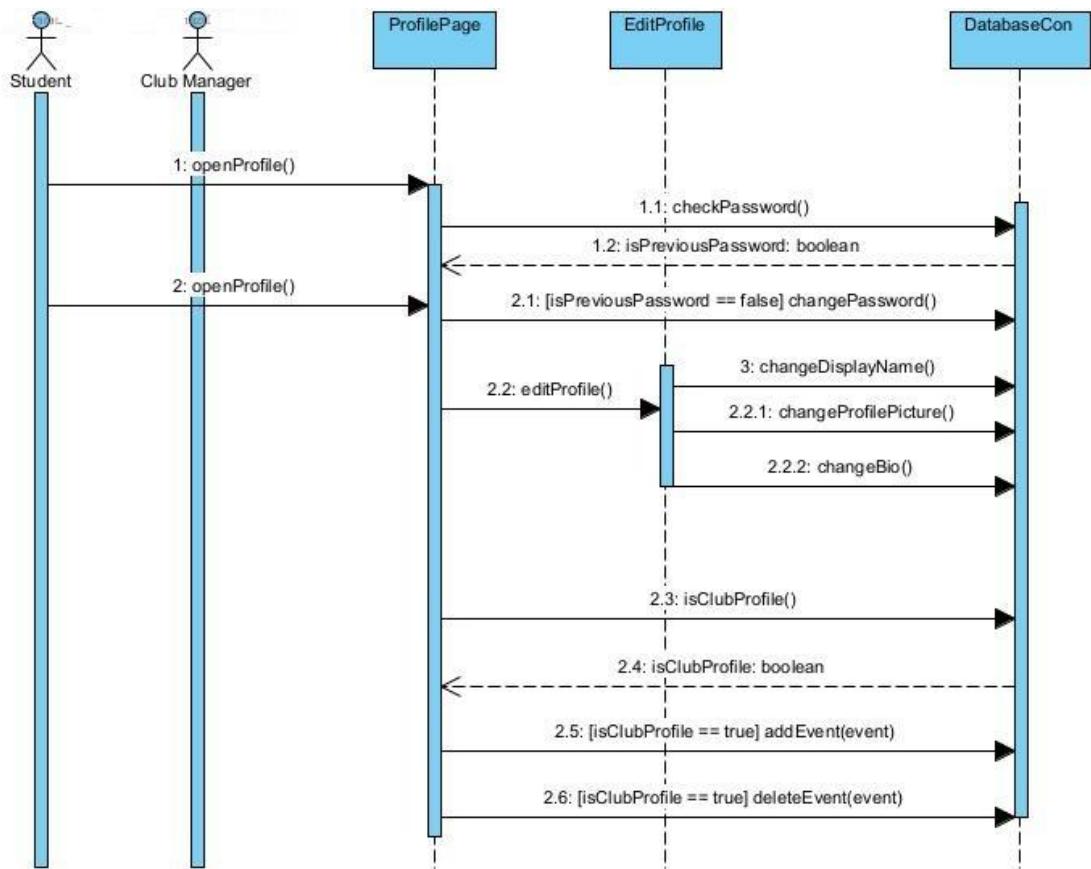


Figure 9: Profile Page Sequence Diagram

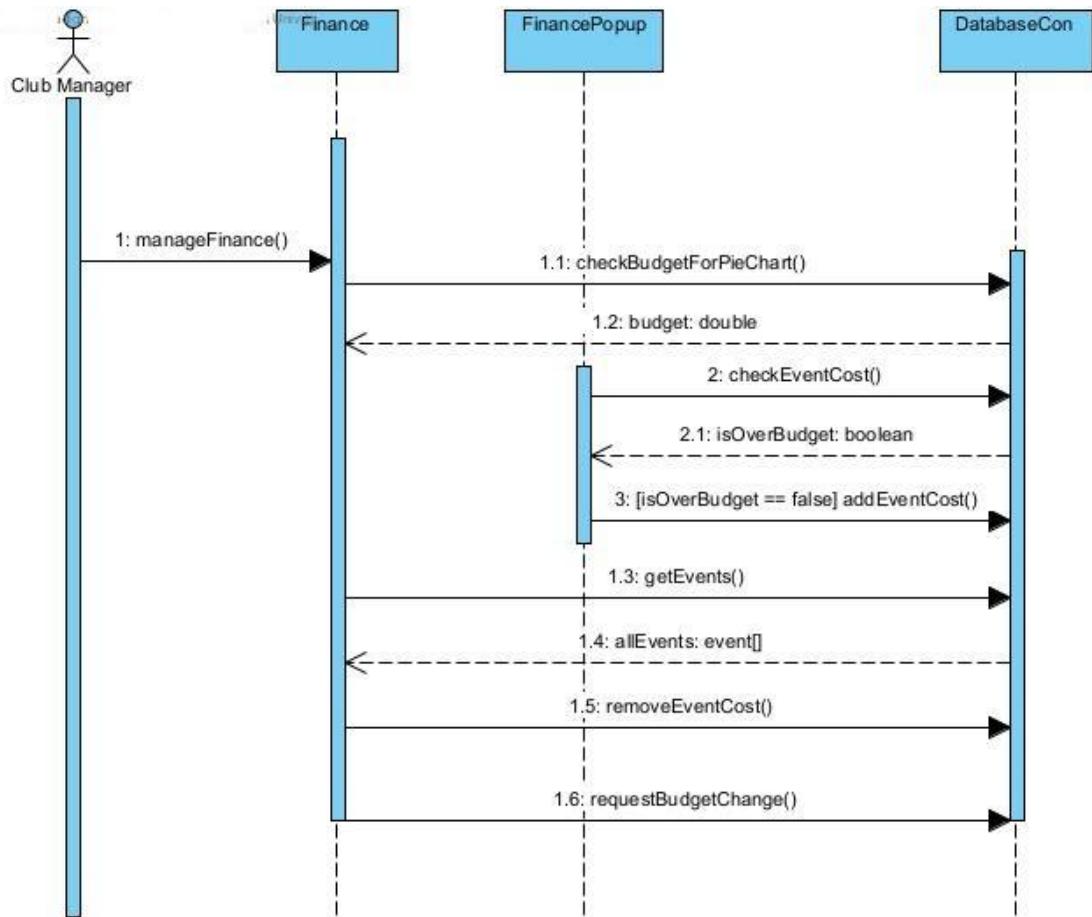


Figure 10: Manage Finance Sequence Diagram

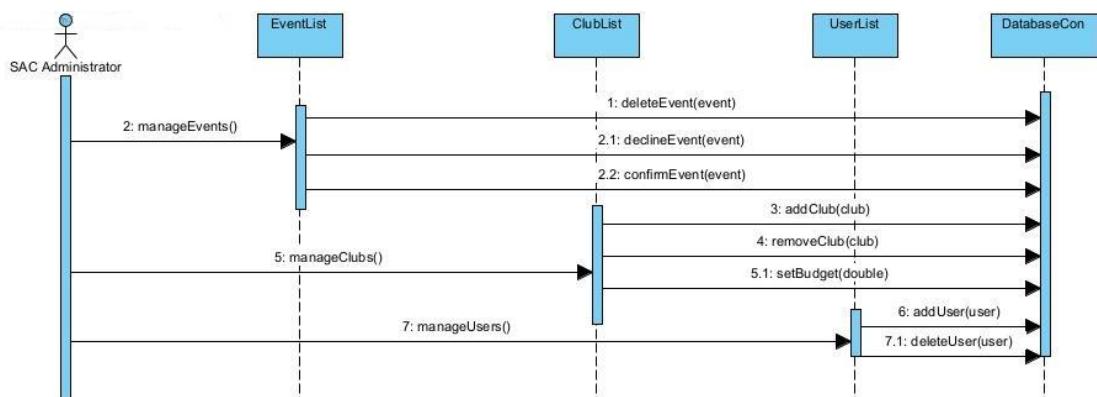


Figure 11: SAC Administrator Sequence Diagram

3.6 State Diagrams

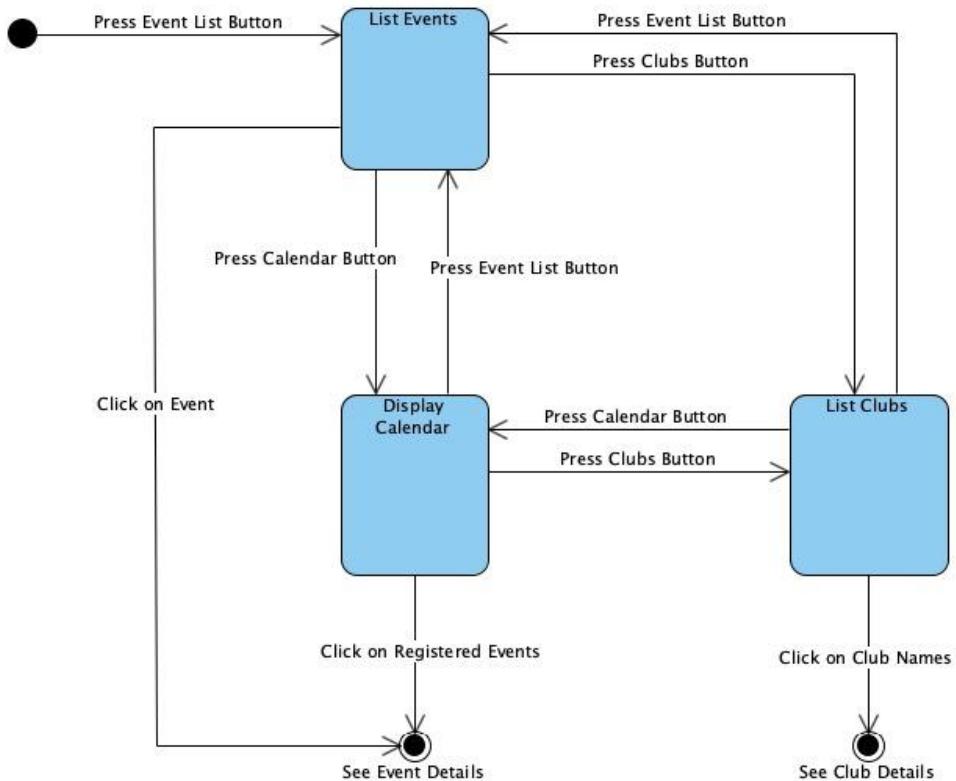


Figure 12: Student State Diagram

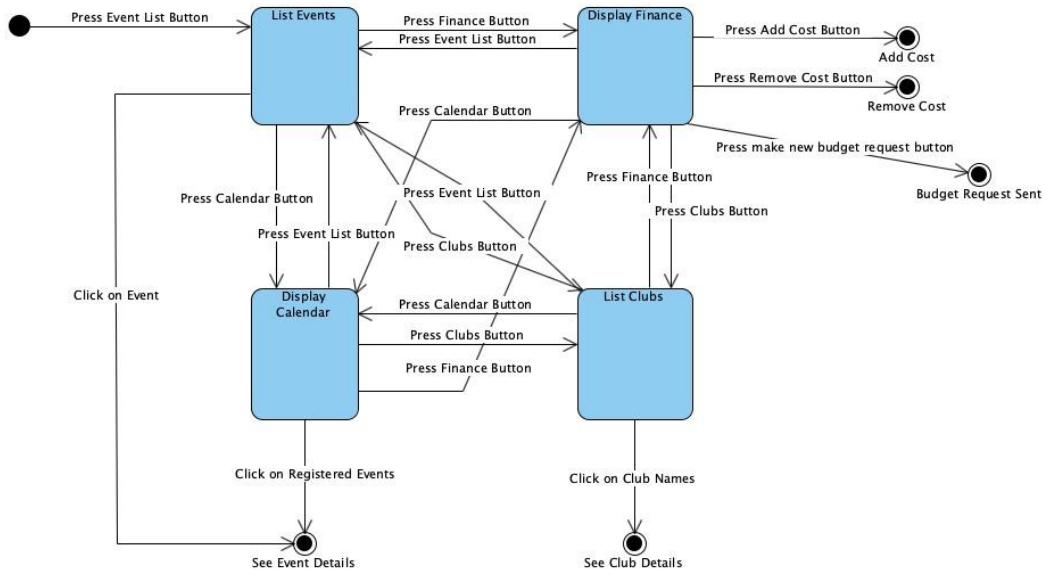


Figure 13: Club Manager State Diagram

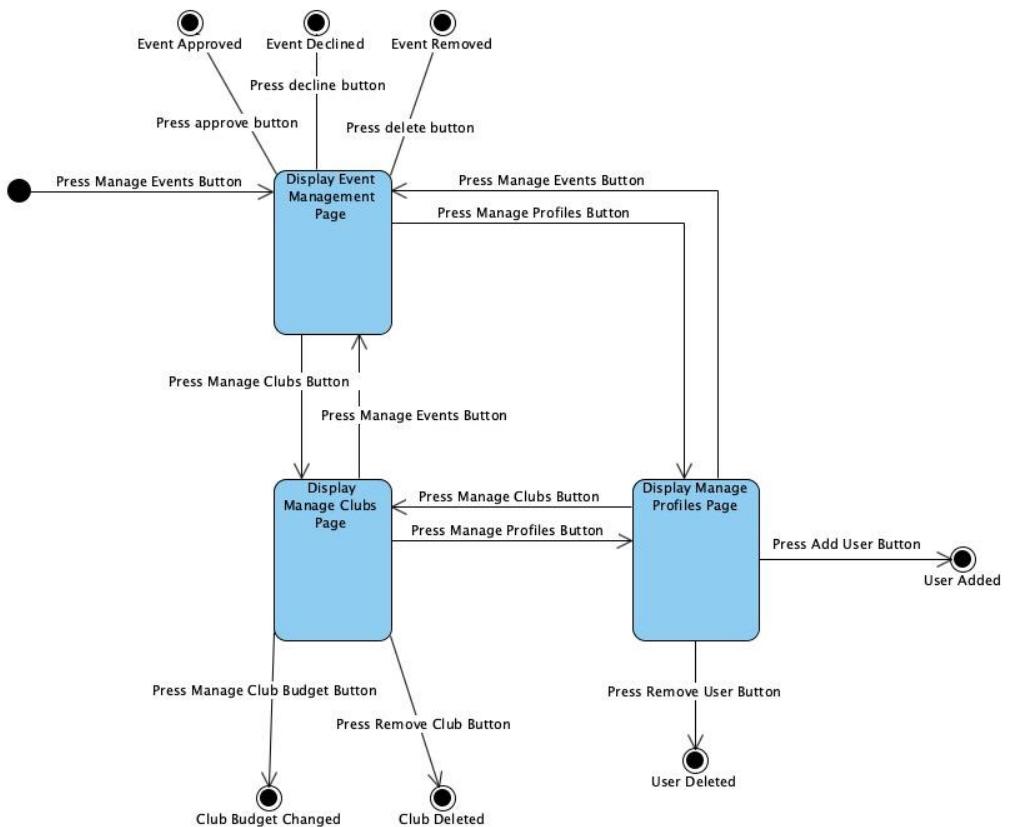
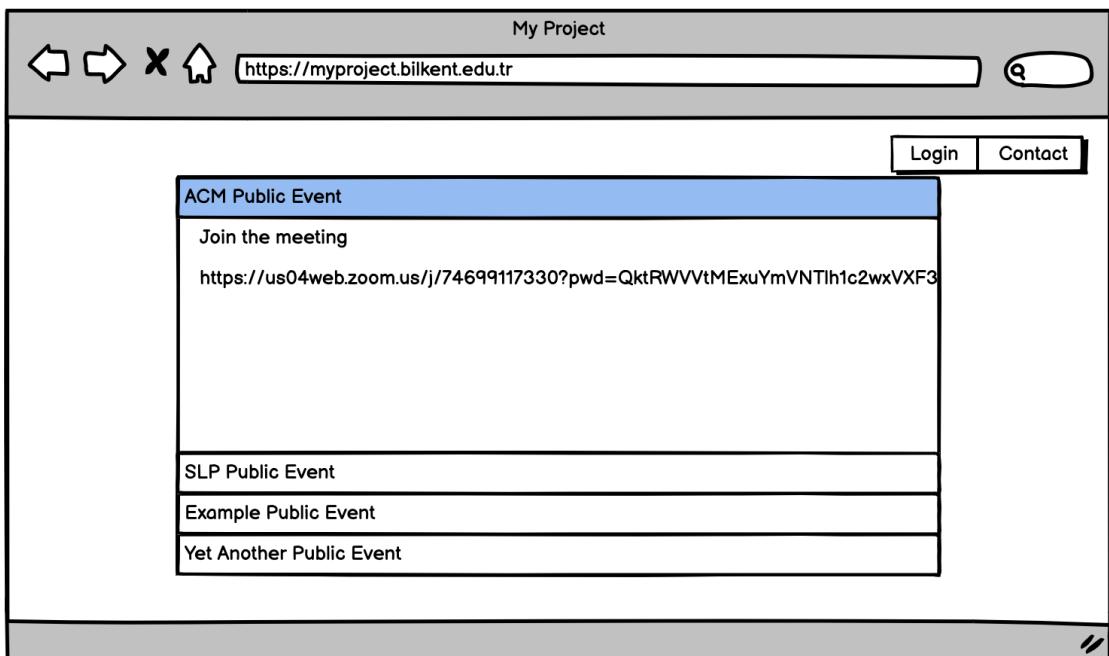


Figure 14: SAC Administrator State Diagram

3.7 Mockup Diagrams (UI Models)



My Project

https://myproject.bilkent.edu.tr

Login Contact

Student Login

Username:

Password:

Club Login

Username:

Password:

My Project

https://myproject.bilkent.edu.tr

Login Contact

Contact with us

If you have any trouble please contact us at example@example.com

Made with <3 and Cola Zero and Filter Coffee

My Project

https://myproject.bilkent.edu.tr

Event List
Calendar
Clubs

Jack London

Event Name	Club	Date	Time Block	Details
Pizza Partisi	ACM	27.10.2021	18.00-20.00	
BEE with Kariyer.net	OR	28.10.2021	17.30-19.30	
Taylan Yıldız ile Söylesi Etkinliği	Atatürkçü Düşünce Topluluğu	28.10.2021	18.30-19.30	
Linux 102	Google Developer Student Club	29.10.2021	17.30-21.00	
Felsefe Atölyesi	Bilkent Felsefe Topluluğu	29.10.2021	17.30-18.30	
Example Event	Example Club	01.11.2021	17.30-18.30	
Example Event	Example Club	01.11.2021	17.30-18.30	
Example Event	Example Club	01.11.2021	17.30-18.30	
Example Event	Example Club	01.11.2021	17.30-18.30	

My Project

<https://myproject.bilkent.edu.tr>

Event List	Calendar	Clubs	Finance	search	ACM	User Icon	Logout
Event Name	Club	Date	Time Block	Details			
Pizza Partisi	ACM	27.10.2021	18.00-20.00	i			
BEE with Kariyer.net	OR	28.10.2021	17.30-19.30	i			
Taylan Yıldız ile Söylesi Etkinliği	Atatürkçü Düşünce Topluluğu	28.10.2021	18.30-19.30	i			
Linux 102	Google Developer Student Club	29.10.2021	17.30-21.00	i			
Felsefe Atölyesi	Bilkent Felsefe Topluluğu	29.10.2021	17.30-18.30	i			
Example Event	Example Club	01.11.2021	17.30-18.30	i			
Example Event	Example Club	01.11.2021	17.30-18.30	i			
Example Event	Example Club	01.11.2021	17.30-18.30	i			
Example Event	Example Club	01.11.2021	17.30-18.30	i			

My Project

<https://myproject.bilkent.edu.tr>

Event List	Calendar	Pizza Partisi		Jack London	User Icon	Logout
Event Name		Pizza Partisi	Leave	Time Block	Details	
Pizza Partisi		V-01		18.00-20.00	i	
BEE with Kariyer.net		27.10.2021		17.30-19.30	i	
Taylan Yıldız ile Söylesi		18.00-20.00		18.30-19.30	i	
Linux 102		ACM		17.30-21.00	i	
Felsefe Atölyesi		The pizza party will be held in V-01. It was only for club members.		17.30-18.30	i	
Example Event				17.30-18.30	i	
Example Event				17.30-18.30	i	
Example Event				17.30-18.30	i	
Example Event				17.30-18.30	i	

My Project

<https://myproject.bilkent.edu.tr>

Pizza Partisi

Event List Calendar

Event Name

Pizza Partisi

BEE with Kariyer.net

Taylan Yıldız ile Söylesi

Linux 102

Felsefe Atölyesi

Example Event

Example Event

Example Event

Example Event

Pizza Partisi
V-01
27.10.2021
18.00-20.00
ACM

The pizza party will be held in V-01. It was only for club members.

Join Available Quota 31/50

Time Block	Details
18.00-20.00	i
17.30-19.30	i
18.30-19.30	i
17.30-21.00	i
17.30-18.30	i

Jack London

Time Block Details

18.00-20.00 i

17.30-19.30 i

18.30-19.30 i

17.30-21.00 i

17.30-18.30 i

17.30-18.30 i

17.30-18.30 i

17.30-18.30 i

My Project

Event List Calendar Clubs Finance ACM

NOVEMBER 2021

S	M	T	W	T	F	S
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	1 Pizza Partisi	2	3	4
5	6	7	8	9	10	11

My Project

Event List Calendar Clubs Jack London

NOVEMBER 2021

S	M	T	W	T	F	S
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	1 Pizza Partisi	2 BEE with Kariyer.net Taylan Yıldız ile Söyleşi	3 Linux 102 Felsefe Atölyesi	4
5	6	7	8	9	10	11

My Project

https://myproject.bilkent.edu.tr

Event List Calendar Clubs

search

Jack London

Operations

Code	Name	Operations
ACM	Association for Computing Machinery	Show More
SLP	Street Lamp Project	Show More
IEEE	IEEE Student Branch	Show More
FLSFT	Philosophy Society	Show More
CAYYK	Tea Club	Show More
RNKDK	Think Colorfully Club	Show More
HYDST	Animals' Friends Club	Show More
BDOST	Outdoor Sports Society	Show More
Example Club Code	Example Club Name	Show More

Example Club Code Example Club Name

My Project

<https://myproject.bilkent.edu.tr>

Event List	Calendar	Clubs	Finance	search	ACM	User Icon	Operations
Code		Name					Show More
ACM		Association for Computing Machinery					Show More
SLP		Street Lamp Project					Show More
IEEE		IEEE Student Branch					Show More
FLSFT		Philosophy Society					Show More
CAYYK		Tea Club					Show More
RNKDK		Think Colorfully Club					Show More
HYDST		Animals' Friends Club					Show More
BDOST		Outdoor Sports Society					Show More
Example Club Code		Example Club Name					Show More

My Project

<https://myproject.bilkent.edu.tr>

Event List	Calendar	IEEE	Jack London	User Icon	Operations		
Code		IEEE Student Branch			Show More		
ACM					Show More		
SLP					Show More		
IEEE		Purpose			Show More		
FLSFT		IEEE, an association dedicated to advancing innovation and technological excellence for the benefit of humanity, is the world's largest technical professional society. It is designed to serve professionals involved in all aspects of the electrical, electronic, and computing fields and related areas of science and technology that underlie modern civilization.			Show More		
CAYYK					Show More		
RNKDK					Show More		
HYDST					Show More		
BDOST					Show More		
Example Club Code		Upcoming Events			Show More		
		Event Name	Date	Time Block	Details		Show More
		Webinar	27.10.2021	18.00-20.00			Show More

My Project

https://myproject.bilkent.edu.tr

Event List Calendar

Budget Change Request

ACM

Event Name: Pizza Party

Current Monthly Budget 25000 TRY

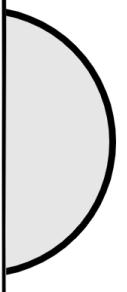
New Monthly Budget

The reason

Submit

Delete Event 

40 %60



Monthly Budget 25000 TRY

Budget Change Request

My Project

https://myproject.bilkent.edu.tr

Event List Calendar

Add Event

ACM

Pizza Partisi

Add Event

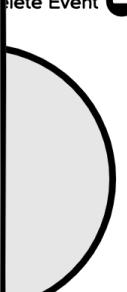
Event Name

Budget

Spent

Delete Event 

40 %60



Monthly Budget 25000 TRY

Budget Change Request

My Project

https://myproject.bilkent.edu.tr

Event List Calendar Clubs search Jack London

Full Name: Jack London
ID: 21700000
Joined Clubs: IEEE, TDP, ACM
Role: Student

Notifications

ACM invited you to join private event

Accept Decline

Jack London About Me

Change Password Edit Profile

My Project

https://myproject.bilkent.edu.tr

Event List Calendar Clubs Finance search ACM

Club Info

Notifications

Jack London accepted to join private event

ACM

Upcoming Events

Create Event + Delete Event -

Event Name	Club	Date	Time Block	More
Pizza Partisi	ACM	27.10.2021	18.00-20.00	i
Example Event	ACM	01.11.2021	17.30-18.30	i
Example Event	ACM	01.11.2021	17.30-18.30	i

Change Password Edit Profile

My Project

https://myproject.bilkent.edu.tr

Event List Calendar

Change Password

ACM

Old Password

New Password

Change Password

to join private event

Delete Event

Block	More
0-20.00	i
0-18.30	i
0-18.30	i

ACM

Change Password

Edit Profile

My Project

https://myproject.bilkent.edu.tr

Event List Calendar

Add Event

ACM

Quota

Event Type

Select Event Type

Public
Campus Only
Private

Upload Photo
(Optional)

Create Event

to join private event

Delete Event

Block	More
0-20.00	i
0-18.30	i
0-18.30	i

ACM

Change Password

Edit Profile

My Project

<https://myproject.bilkent.edu.tr>

Event List Calendar

Add Event

Club Info

ACM

Change Password

Edit Profile

Event Name: []

Date: []

Time: []

Location: []

Description: []

to join private event

Delete Event []

Time Block More

0-20.00	[i]
0-18.30	[i]
0-18.30	[i]

My Project

<https://myproject.bilkent.edu.tr>

Event List Calendar

Club Info

ACM

Change Password

Edit Profile

Delete Pizza Partisi

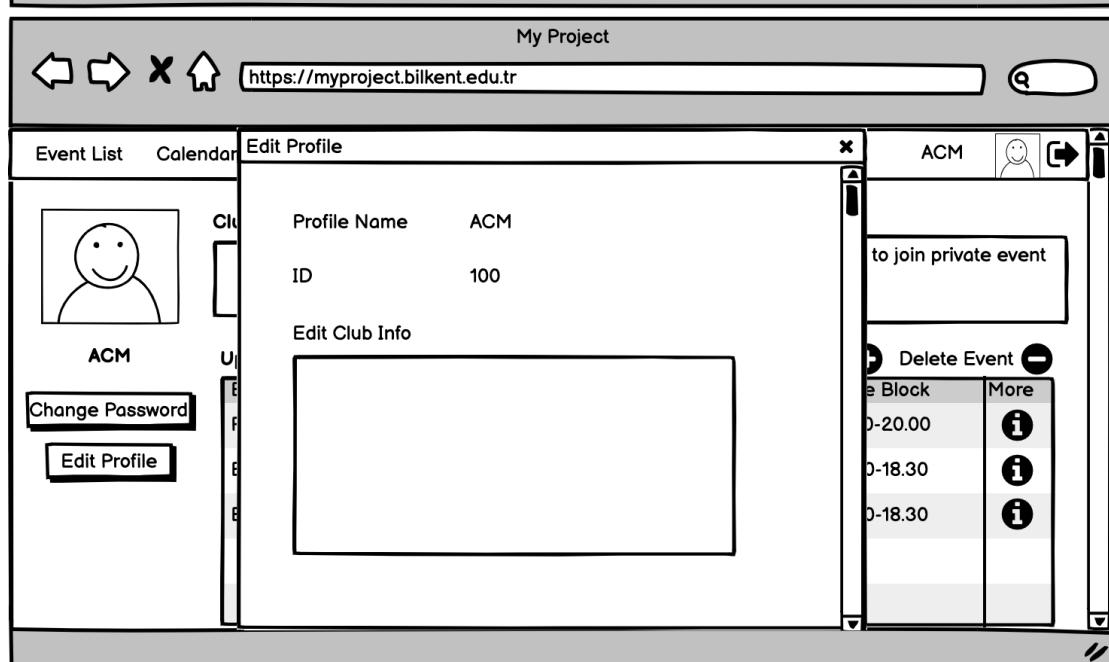
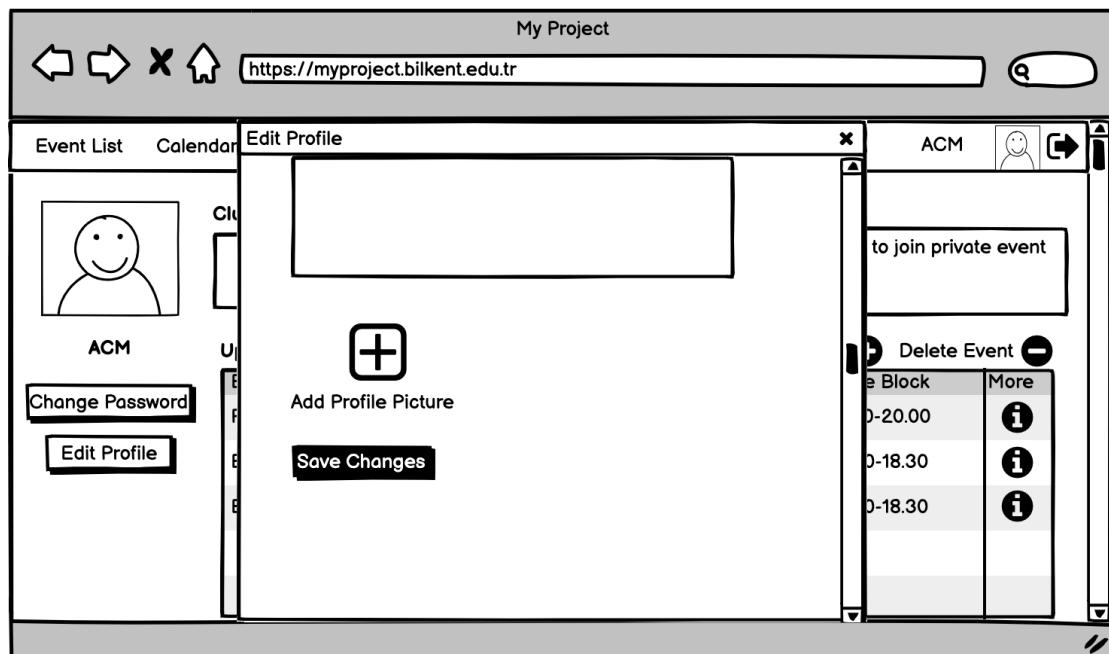
Event Name	Club	Date	Time Block	Delete
Pizza Partisi	ACM	27.10.2	18.00-20.00	[]
Example Event	ACM	01.11.20	17.30-18.30	[]
Example Event	ACM	01.11.20	17.30-18.30	[]

Join private event

Delete Event []

Time Block More

0.00	[i]
30	[i]
30	[i]



My Project

https://myproject.bilkent.edu.tr

Event List Calendar

Jack London

Fu
ID:
Join
Ro

Old Password

New Password

Change Password

Edit Profile

Change Password

Jack London

Ab

My Project

https://myproject.bilkent.edu.tr

Event List Calendar

Jack London

Fu
ID:
Join
Ro

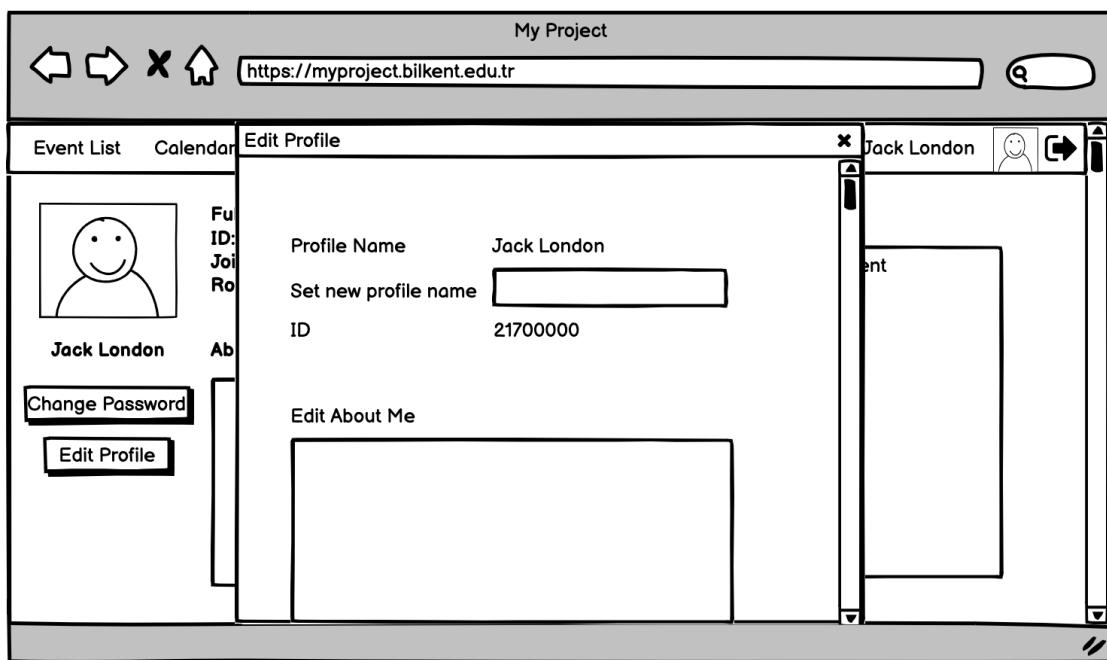
Add

Add Profile Picture

Save Changes

Jack London

Ab



My Project

https://myproject.bilkent.edu.tr

Manage Events Admin

Add Profile

Code: ACM, SLP, IEEE, FLSFT, CAYYK, RNKDK, HYDST, BDOST, Example Club Code

Add Profile

Profile Name:

ID:

Password:

User Type: Please Select User Type

Operations

Admin [User Icon] [Logout] [Print]

Operations grid:

\$1	-
\$1	-
\$1	-
\$1	-
\$1	-
\$1	-
\$1	-
\$1	-

My Project

https://myproject.bilkent.edu.tr

Manage Events Admin

Manage Budget

Code: ACM, SLP, IEEE, FLSFT, CAYYK, RNKDK, HYDST, BDOST, Example Club Code

ACM

Set Budget

Current Monthly Budget: 25000 TRY

New Monthly Budget:

The reason:

Operations

Operations grid:

\$1	-
\$1	-
\$1	-
\$1	-
\$1	-
\$1	-
\$1	-
\$1	-

My Project

<https://myproject.bilkent.edu.tr>

Manage Events	Manage Clubs	Manage Profiles	Admin
Code	Name	Operations	
ACM	Association for Computing Machinery		
SLP	Street Lamp Project		
IEEE	IEEE Student Branch		
FLSFT	Philosophy Society		
CAYYK	Tea Club		
RNKDK	Think Colorfully Club		
HYDST	Animals' Friends Club		
BDOST	Outdoor Sports Society		
Example Club Code	Example Club Name		

My Project

<https://myproject.bilkent.edu.tr>

Manage Events	Manage Clubs	Manage Profiles	Admin	
Event Name	Club	Date	Time Block	Operation
Pizza Partisi	ACM	27.10.2021	18.00-20.00	
BEE with Kariyer.net	OR	28.10.2021	17.30-19.30	
Taylan Yıldız ile Söylesi Etkinliği	Atatürkü Düşünce Topluluğu	28.10.2021	18.30-19.30	
Linux 102	Google Developer Student Club	29.10.2021	17.30-21.00	
Felsefe Atölyesi	Bilkent Felsefe Topluluğu	29.10.2021	17.30-18.30	
Example Event	Example Club	01.11.2021	17.30-18.30	
Example Event	Example Club	01.11.2021	17.30-18.30	
Example Event	Example Club	01.11.2021	17.30-18.30	
Example Event	Example Club	01.11.2021	17.30-18.30	

My Project

<https://myproject.bilkent.edu.tr>

Manage Events	Add Profile	Admin
Profile Name Jack London Example Student Name	 <input type="button" value="Add Profile"/> Profile Name: <input type="text"/> ID: <input type="text"/> Password: <input type="text"/> User Type: <input type="button" value="Please Select User Type"/> <div style="border: 1px solid black; padding: 2px; display: inline-block;"> Admin Student Club </div>	Operations        

My Project

<https://myproject.bilkent.edu.tr>

Manage Events	Manage Clubs	Manage Profiles	 <input type="text" value="search"/>	Admin
Profile Name Jack London Example Student Name		ID 21700000 21X00000		Operations        

Glossary

- SAC: Student Activities Center

References

- [1] Object-Oriented Software Engineering, Using UML, Patterns, and Java, 2nd Edition, by Bernd Bruegge and Allen H. Dutoit, Prentice-Hall, 2004, ISBN: 0-13-047110-0.