



CS319 Term Project

Project short-name: BilBoard

Section 3

Group 3a

Project Analysis Report

Project Group Member Names:

İlke Doğan (21702215)

Hacı Çakın (21802641)

Metehan Saçakçı (21802788)

Muhammet Abdullah Koç (21802832)

Aslı Dinç (21802527)

Instructor: Eray Tüzün

Table of Contents

1 Introduction	4
2 Proposed System	4
2.1 Overview	4
2.2 Functional Requirements	4
2.2.1 Login	5
2.2.2 Register	5
2.2.5 Seeing the Details of an Event	5
2.2.6 Participating in Survey	6
2.2.7 Searching Event	6
2.2.8 Taking Attendance with Using Special Code	6
2.2.9 Managing the User Profile	6
2.2.10 Managing and Viewing the Club Profile	6
2.2.12 Adding Event	7
2.2.13 Editing Event	7
2.2.14 Adding Survey	7
2.2.15 Adjusting Club Hierarchy	8
2.2.16 Managing Club Sponsors	8
2.2.17 Giving Feedback	8
2.2.18 Arrange Building/Class Condition	8
2.2.19 Viewing Club Member	8
2.2.20 Assigning Specific Roles	8
2.3 Non-functional Requirements	9
2.3.1 User-Friendly Interface	9
2.3.2 Security	9
2.3.3 Maintainability	9
2.3.4 Reliability	9
2.3.4 Scalability	10
2.4 Pseudo Requirements	10
2.5 System Models	10
2.5.1 Scenarios	10
2.5.1.1 Add Event Scenario	10
2.5.1.2 Taking Attendance Scenario	11
2.5.1.3 Adjust Club Hierarchy Scenario	11
2.5.1.4 Add Survey Scenario	12
2.5.2 Use-Case Model	13
2.5.3 Object and Class Model	15
2.5.4 Dynamic Models	19
2.5.4.1 Sequence Diagram	19

2.5.4.2 State Diagrams	21
2.5.4.3 Activity Diagrams	23
2.5.5 User Interface	25
About Bilboard Page	26
Login and Register Page	26
Main Page	27
Calendar Page	28
Hovering on an Event	28
Event Page Seen by Students	29
Club Page from Student User's Perspective	30
Leave Feedback Dialog	31
Surveys Page	31
Attend Event Dialog	32
User Profile Settings Page	33
Club General Settings Page	34
Add Event Page	35
Feedbacks Page	36
Survey Settings Page	37
Club Hierarchy Page	38
Club Sponsor(s) Page	39
Administrative Assistant Page	40
Event Page Seen by Board Members	41
Admin Add Club Screen	42
Admin Manage Club Screen	42
3 Glossary	43
4 Improvement Summary	44
4.1 Introduction	44
4.2 Proposed System - Functional Requirements	44
4.3 Non-functional Requirements - Reliability	44
4.4 Non-functional Requirements - Scenarios	44
4.5 Use Case Diagram	44
4.6 Class Diagram	45
4.7 Sequence Diagram	45
4.8 State Diagram	45
4.9 Activity Diagram	45
4.10 User Interface	45
5 References	46

1 Introduction

Our application is a cross-platform website application project that is designed for student-centered club activities. This project will be utilized by admin, club presidents, club board members, club members, active members, advisors, administrative assistant, and students & academic personals [non-member]. This application offers various options for the users regarding club events & meetings, club management, surveys, daily event notifications.

Board members and president can add an event to the system with a specified number of students

Students can follow the clubs over the application, and turn into club members;

1. Club members can activate the event notifications
2. Club members can join the WhatsApp groups of the clubs by link
3. Club members can be informed about event time and place and make reservations (Face to face and Online Meetings are available)
4. Club members can participate in surveys

Board members and president can arrange surveys

Board members and president can take attendance with using a special code arranged for specified each event

Based on the attendance amount, club members can turn into active members and give a vote for the election

2 Proposed System

2.1 Overview

We firstly discussed on which platform we would like to develop our project. It should be underlined that we could not come out with a decision for this question during our previous meeting, however, after the decision made by instructors regarding web-based projects, we started to draw the baseline of the project for web application with suitable technologies.

2.2 Functional Requirements

There are 8 numbers of user types that are lined up in this heading in a hierarchical order: Admin, Club Presidents, Club Board Members, Club Members, Active Members, Advisors, Administrative Assistant, and Students & Academic Personals[Non-Member].

Admin is a special position that manages the whole app and is able to give auth to club presidents.

Club President will have the described abilities above and additionally be able to edit the board members of the club.

Advisor is a role that each club has one. It should be seen that every club in Bilkent University must be assigned to an academic personnel of the university. With our application each club's advisor's can be formally displayed and can be seen.

Club Board Members will have the described abilities above and additionally can share, remove and edit events and surveys of clubs and also can take the attendance per event via shared code.

Active Members are able to join the voting activity for both survey and the election which is realized for board members of club election.

Club Members will have the described abilities above and additionally directly see the events of the joined club on the home main thread.

Students and Academics[Non-Member] will be able to see and attend events of clubs and become members of the clubs.

Administrative Assistant is a position that arranges the face-to-face event locations and times.

2.2.1 Login

On the login page, the user will provide the login information to the application in order to join the program. After successful login, the user will be directed to the home page.

2.2.2 Register

On the register page, potential users will specify the required information into the application to register. After a successful register operation, a confirmation mail is sent to the specified mail. After the user confirms himself/herself, he/she can login to the application.

2.2.5 Seeing the Details of an Event

The user will be able to see the events that belong to the clubs which the user has joined as a member and the user will be able to reach all of the details about the event provided by the club. The users can have information about event name, image, date, location, and description. Also, whether GE250/251

points will be given can be seen. In addition, users can see a discover section that includes some events from clubs that users did not join before. Thus, a user can discover other clubs and their activities.

2.2.6 Participating in Survey

In the Survey page, users can see the surveys opened by student clubs. Users can vote in the surveys and submit their votes. However, being able to vote in the survey requires being an active member.

2.2.7 Searching Event

All upcoming events and clubs can be found on the search button. If the user wants to see other events besides the events of the clubs he follows, he can use this button. However, some events are exclusive to active members only by clubs. Therefore, they will not be allowed to participate in these events.

2.2.8 Taking Attendance with Using Special Code

Currently, attendance of the events is usually taken with collecting signatures of the attendees. However, in our application, board members will be able to generate attendance codes for each event and attendees will type these codes in their account. In that way their attendance will be formally taken with our application.

2.2.9 Managing the User Profile

Users will be able to see their general account information. If the user has a rank such as “Club Board Member” or “Club President”, they will be able to directly pass the related club’s page from their profile page. Also, the users will be able to edit/change their user preferences.

2.2.10 Managing and Viewing the Club Profile

Club Profile is the main page of the clubs seen by the students. On this page, all event posts can be seen. In addition, students can join the Whatsapp group of the club, or go to the Instagram account of the club. Moreover, students can become a member of the club by clicking the “Become Member”

button. Finally, students can give private feedback about the club and the events of the specified club. Participants also can ask questions regarding the events.

Board members can edit the profile preferences of the club. They can basically change the profile photo of the club, Whatsapp link, or Instagram link. In addition, board members can edit the attributes of the events.

2.2.12 Adding Event

Our application has a feature for adding an event. A board member uploads a photo of the event, adds event name and description, sets number of participants and sets the event time and place. If the event is online and the user selects “list event after creation”, the event can be seen by the students after the creation of the event. Otherwise, the event becomes unlisted and board members can list the event. If the event is face-to-face, the information about the event is sent to the administrative assistant. If the administrative assistant confirms the place and time, the event becomes unlisted. If the place is used for another event, the assistant sends the event back to the board members and requests change on the place.

2.2.13 Editing Event

Board members also can edit the event details. If board members set the new information and click the “Submit” button, the new information is processed and can be seen by the students. In addition, if an event’s time and place are not appropriate, feedback is sent to board members by the administrative assistant. In this case, board members should edit the event time and place, and submit again. Moreover, board members can set the visibility of the event. If the event is not visible, the event cannot be seen by students.

2.2.14 Adding Survey

Board members can manage surveys on the survey page. So as to add a survey; survey question, survey choices and end date should be specified. When end time comes, the survey automatically removes from students’ survey pages. Board members can also see the results of continuing surveys and end the surveys although the end date has not come.

2.2.15 Adjusting Club Hierarchy

The Club Hierarchy page is accessible by the club president. On this page, the club president can add or remove board members. In order to add a board member, the student's name and ID should be specified.

2.2.16 Managing Club Sponsors

Club sponsors can be added or removed by the Club Sponsors page. In order to add a sponsor; sponsor name, sponsor logo, amount of donation, and sponsorship type (platinum, gold, silver, bronze) should be specified. To remove a sponsor from the sponsor list, clicking the delete button at the right of the sponsor is enough.

2.2.17 Giving Feedback

In the Feedback page, board members can see the feedback given for the club. These feedbacks can only be seen by the board members and club presidents. Feedbacks can be deleted.

2.2.18 Arrange Building/Class Condition

A page designed for administrative assistants, helps add/remove buildings and classes located in Bilkent University.

2.2.19 Viewing Club Member

In the club management page, advisors, club president, admin are able to view the list of club members enrolled in a specified club.

2.2.20 Assigning Specific Roles

In the admin screen, admin can assign advisor and president to a club. In the club hierarchy page, the club president is able to arrange hierarchy in the club.

2.3 Non-functional Requirements

2.3.1 User-Friendly Interface

User-friendly interface will be ensured. This will be achieved by the following: Font size will not be lower than 12. For the navigation, an app bar will be used because it is the common practise. All the related features will be under the suitable sections. Moreover, if users want to search anything, this app bar will be accessible for everywhere in the site. While waiting for the requests' responses, users will always be notified about the process. If something went wrong from the backend side(anything that has status code 4XX-5XX), users will be informed about that in the user interface. Since our application will be a single page application, it will be much more responsive while users browse our application.

2.3.2 Security

In terms of security, users who are in Bilkent will be eligible to sign up for this application. To provide this, only bilkent emails will be accepted. Security passwords of users will be encrypted. All the requests (except sign in, sign out, forget password requests) which will be made by client side, will be authorized by using JWT(JSON Web Token). These JWTs will be created when users sign in, and will expire in 1 hour. In addition to this, by using CORS(Cross-Origin Resource Sharing), all the requests which are not made from the frontend part will be responded to by CORS error. All the passwords will be at least 8 characters.

2.3.3 Maintainability

For the maintainability, since the application will be done by using OOP(Object Oriented Programming), adding new features or bug fixes will not change the legacy of the code. System will become bigger but code fragments will not grow. In addition to this, the frontend part will be constructed as component-based.

2.3.4 Reliability

One of the most crucial parts of non-functional requirements is reliability for the users. All the possible bad request responses will be covered for the client side.

2.3.4 Scalability

For scalability, the size of all responses will not exceed 50 kb. If the requested data will grow in time, growing part of the response should be obtained by pagination. This makes it possible to preserve the client side from huge increments in response sizes.

2.4 Pseudo Requirements

These are the pseudo requirements of the application.

Implementation language will be Java and Java 8 will be used.

For the mailing, google mail will be opened and Mails will be sent through this mail by using IMAP access.

Model Mapper library will be used to provide object transformation between the business layers.

In the database, MySQL which is relational will be used.

For the client side, React which is one of the biggest JavaScript frameworks will be used.

For tracing and controlling changes, GitHub will be used as a version control system.

Backend of the system will be deployed to AWS(Amazon Web Service)'s Elastic Beanstalk. It will work on the Amazon Corretto 8 machine. Database of the system will also be placed on the same machine.

Build of the backend part will be taken as a Jar file.

Frontend part of the system will be deployed to Google Firebase Hosting service.

Build of the frontend part will be taken as a single page build.

Application's target will be mainly Chrome, Safari, Firefox and Internet Explorer.

2.5 System Models

2.5.1 Scenarios

2.5.1.1 Add Event Scenario

- 1.Name: Add Event
- 2.Participating Actor: Club Board Member, Administrative Assistant
- 3.Entry Condition: Being Board Member/Club President

- 4.Exit Condition: Event is uploaded to the system or event add request is denied by the Administrative Assistant.
- 5.Flow of Events:
 1. Club Board Member clicks the “Add Event” button which is located in the Club Profile Page.
 2. Club Board Member needs to enter all of the following information: Title of the Event, Description of the Event, Date of the Event, Location of the Event (if it is an online event, meeting link will be passed; else room and the building information will be passed), maximum number of participant capacity.
 - the artifact’s type, pick contributors, add notes and categorize the artifact’s purpose.
 3. Club Board Member clicks the “Add” button. If the event is online, the process will be completed. However, if it is a face to face event, the Administrative Assistant must approve the room request.

2.5.1.2 Taking Attendance Scenario

- 1.Name: Taking Attendance
- 2.Participating Actor: Club Board Member, Attendees (they can be “Non-Member”, “Club Member, or Active Member”)
- 3.Entry Condition: Being Club Board Member. Maximum attendees capacity should not be exceeded
- 4.Exit Condition: Attendance data is sent to the system.
- 5.Flow of Events:
 1. Club Board Member activates the attendance code which is special for that event and shares them with the attendees via using projector or whiteboard.
 2. In the limited amount of time, attendees must enter the shared code to the application.
 3. If the attendee’s code is correct, his/her attendance will be valid.

2.5.1.3 Adjust Club Hierarchy Scenario

- 1.Name: Adjust Club Hierarchy
- 2.Participating Actor: Club President
- 3.Entry Condition: None
- 4.Exit Condition: New Club Hierarchy data will be sent to the system.
- 5.Flow of Events:
 1. Club President clicks the “Club Hierarchy” button which is located in the Club Profile page.
 2. If the Club President would like to add a new Club Board Member, he/she needs to type the new Club Board Member’s ID and the Name and click the

“Add” button. Else, he would like to delete an existing Club Board Member, he should select the corresponding trash icon.

3. Adding/Deleting Club Board Member(s) will be completed.

2.5.1.4 Add Survey Scenario

- 1. Name: Add Survey
- 2. Participating Actor: Club Board Member
- 3. Entry Condition: Being Board Member/Advisor
- 4. Exit Condition: New Survey data will be sent to the system.
- 5. Flow of Events:
 1. Club Board Member clicks the “Surveys” button which is located in the Club Profile page.
 2. In this page, Club Board Members must specify the survey question4. In order to add new choices, he/she can use the “Add Choice” button.
 3. The end date of the survey must be selected.
 4. After every needed information is specified, Club Board Members can press the “Add Survey” button to publish it.

2.5.2 Use-Case Model

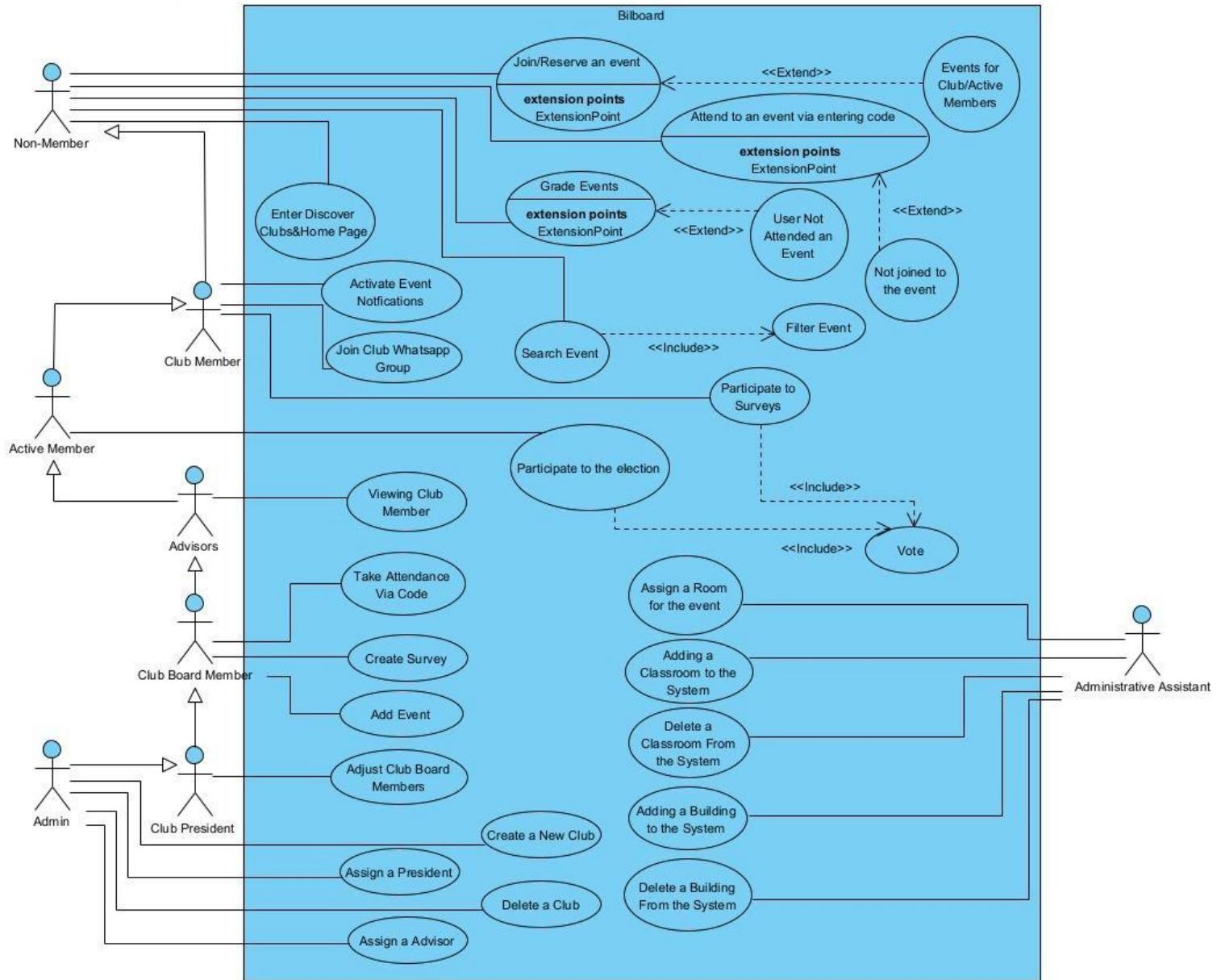


Fig.1. Use Case Diagram.

In the user case diagram above, it should be seen that there is a user hierarchy for “Non-member”, “Club Member”, “Active Member”, “Club Board Member”, “Advisor”, “Club President”, and “Admin” roles. Each role is enclosed by the other one who is below them according to the hierarchy order. Additionally, “Administrative Assistant” is added as a system from Bilkent University .

Non-Member: A person from Bilkent University who is not participating in any club can join the events of the clubs if the owner club the event will not put a prerequisite role for that event. Also if the user joins an event, during the event his/her attendance can be taken formally by using the “entering code” procedure. Also after the event is finished, he/she will be able to grade the event out of 5. Last but not least, while they are searching events, they will be able to filter in the events by their attributes.

Club Member: First of all the “Club Member” role will have every feature that is available for the “Non-Member” role. However, because they are members of a specific club, they can turn on the club notifications to get information about incoming events automatically. Also, if their club shares a Whatsapp group link, they will be able to see the link. Furthermore, they also will be able to join the surveys of their clubs.

Active Member: After a “Club Member” attends a specific amount of club events, they will become an “Active Member” and they will have every feature that is available for the “Club Member” role. Other than that an “Active Member” will be able to participate in Club President Election and Survey Elections.

Club Board Member: “Club Board Member” role will have every feature that is available for the “Active Member” role. Furthermore, they will be able to create events and surveys. They also will be able to generate attendance codes and take formal attendance of the club events.

Advisor: “Advisor” role is used to show the academic personals that are connected to one of the clubs of Bilkent University and Advisors can view the club member list.

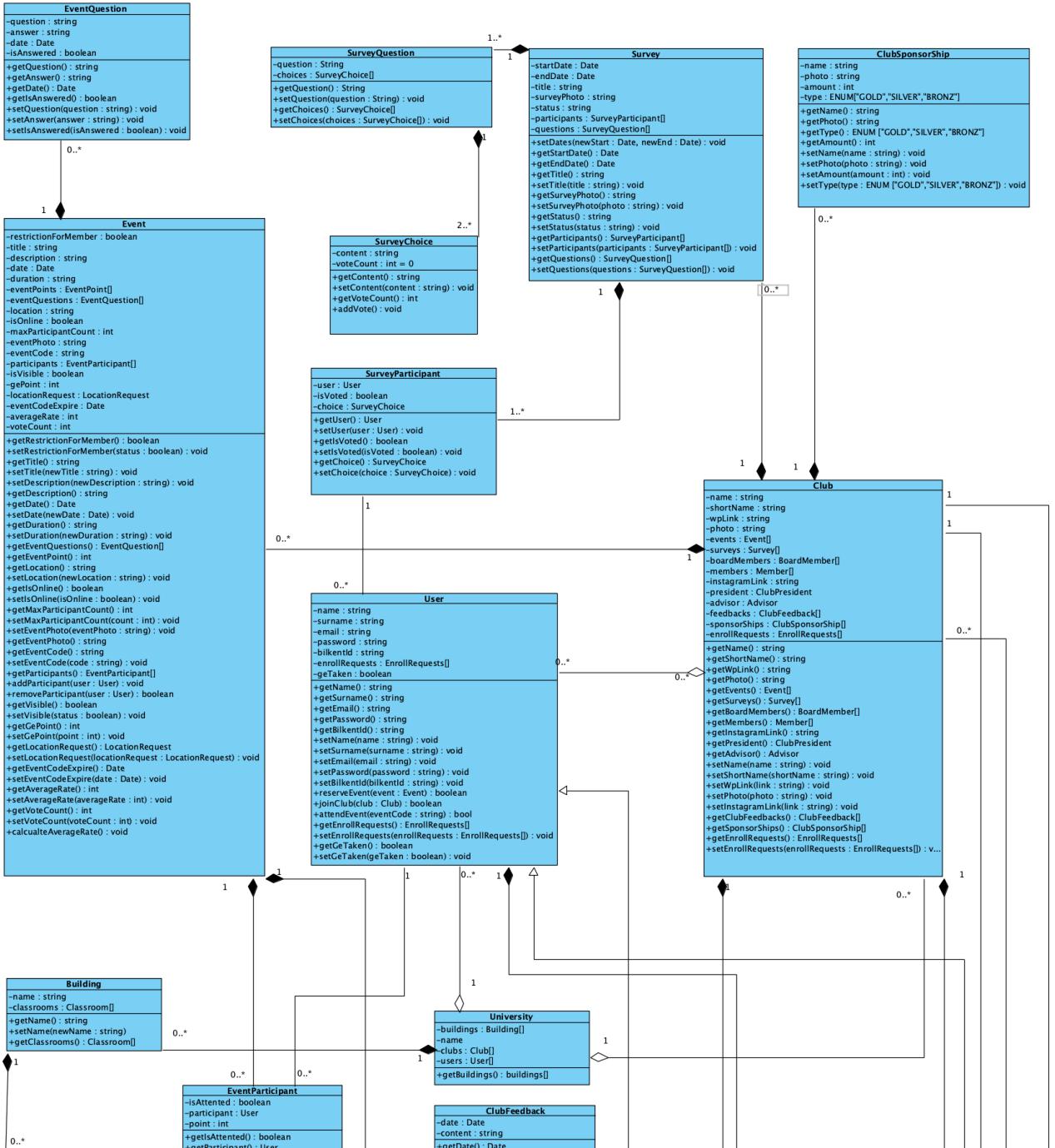
Club President: “Club President” role will have every feature that is available for the “Club Board Member” role. As an extra he/she will be able to adjust the club hierarchy and club board members of the club.

Admin: “Admin” role will be the most authorized role in our application. One of the critical mission of the admins is to assign selected presidents to their clubs and assign Advisors to related club. Additionally, Admin can create/delete a club, and also assign a President/Advisor to a specified club.

Administrative Assistant: This role is one of the special role that is outside the user role hierarchy. This role basically will be used to adjust rooms for

planned events. Administrative Assistant can add/delete a classroom and building to the system.

2.5.3 Object and Class Model



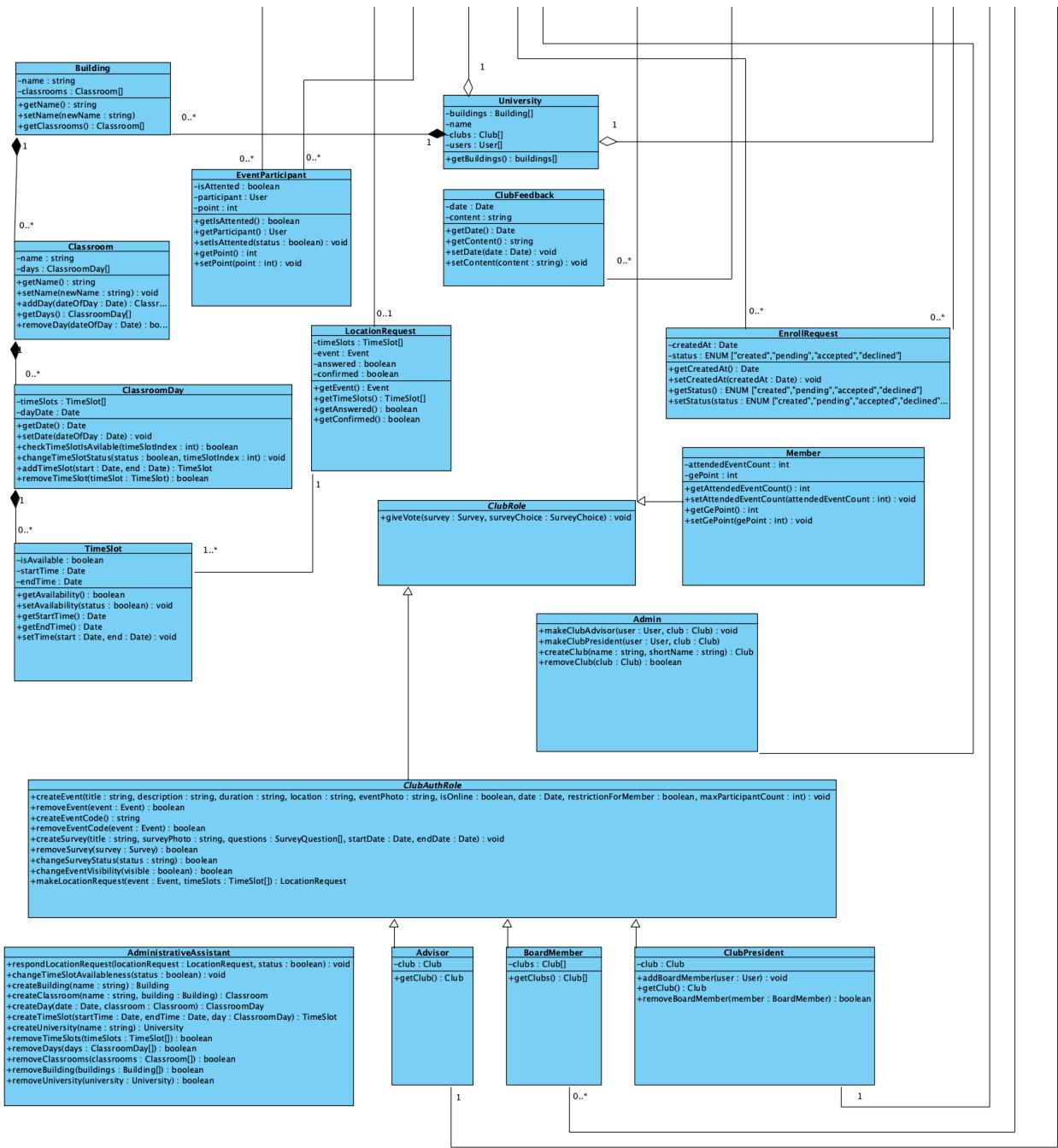


Fig. 2. Class Diagram.

The class models of this application are given above. Application will have 26 classes. These classes are demonstrations of all objects in the application. In this diagram, all relations between the objects can be viewed.

User Class: User class is the base of all user role objects which means it is the superclasses of all user-relative classes. All the users in the application will have properties and abilities which are determined in the User class. The

roles that are students and academics can be constructed directly using this class in the application.

AdministrativeAssistant Class: This class is for the initialisation of the administrative assistant role. Administrative assistants have a responsibility to schedule event places.

ClubRole Class: This is an abstract class to differentiate other users from club users. This abstract class gives the ability to vote in club surveys to all club roles. It is the super classes of the Member Class and ClubAuthRole Abstract Class.

Member Class: This class is the subclass of the ClubRole Abstract Class. All the members of the clubs will be constructed from this class.

ClubAuthRole Class: This is another abstract class that the application has. This abstract class is to distinguish club authorization roles to other roles. It is the subclass of the ClubRole and superclass of the ClubAdvisor, Club President and BoardMember Classes. This class provides access to edit, create, delete events and surveys of the club.

Advisor Class: Each club has an advisor. Advisors have abilities board members have. This class is for the construction of Advisor users.

BoardMember Class: This class is for the formation of the board members of the club. It is a subclass of the ClubAuthRole Abstract Class.

ClubPresident Class: This class is for creating club president objects in the application. Club presidents have the ability to add/remove board members from the club in addition to ClubAuthRole Abstract Class abilities.

Admin Class: This class gives access to determine club advisors, club presidents and administrative assistants in the application.

Club Class: This class is for the construction of the club objects. Under the club, there can be multiple events, members, surveys, board members, and sponsors.

EnrollRequest Class: This class traces enrollment requests that come to the club.

Survey Class: This class is for creating surveys of the clubs. It can have 2 to n options.

SurveyQuestion Class: This class is to create survey options under the survey object. It consists of choices and question content.

SurveyChoice Class: This class is the object of the question choice. It consists of the vote count and choice content.

SurveyParticipant Class: This class is for tracing survey participation. It consists of the isVoted boolean and choice selected by user.

Event Class: The event class is the object of the club events that consist of lots of data. Events will be customizable in this application. It has other objects like Event Question, Event Point, Event Participant, etc.

EventQuestion Class: The event question is for users to ask their event-related questions before the event. After the questions are answered they will be visible to all.

EventParticipant Class: In order to determine the activeness of members of the clubs, each event participant should be followed. This is why the application needs an Event Participant object.

ClubFeedback Class: Each club will have anonymous feedback to give information to possible members. This class is for creating these feedback objects.

University Class: All these objects are the leaves of the University objects. Classes, clubs, etc. are in the University. This class is for the construction of University objects.

Building Class: Each classroom is in a building therefore this class is for creating buildings. This object will have multiple classes and a name.

Classroom Class: Classroom Class is to trace the daily status of the classrooms for the events.

ClassroomDay Class: ClassroomDay Class is to follow daily time slots for each class.

Timeslot Class: The timeslot Class is for the construction of timeslots. These time slots can be reserved for events.

LocationRequest Class: For face-to-face events, clubs have to reserve time slots. For this purpose, location requests can be created. If administrative assistants accept, time slots will be reserved for that event. In order to trace these requests, LocationRequest Class can be used to create these objects.

ClubSponsor Class: Clubs may have sponsorship agreements. To construct these agreements in object base, ClubSponsor Class is needed.

2.5.4 Dynamic Models

2.5.4.1 Sequence Diagram

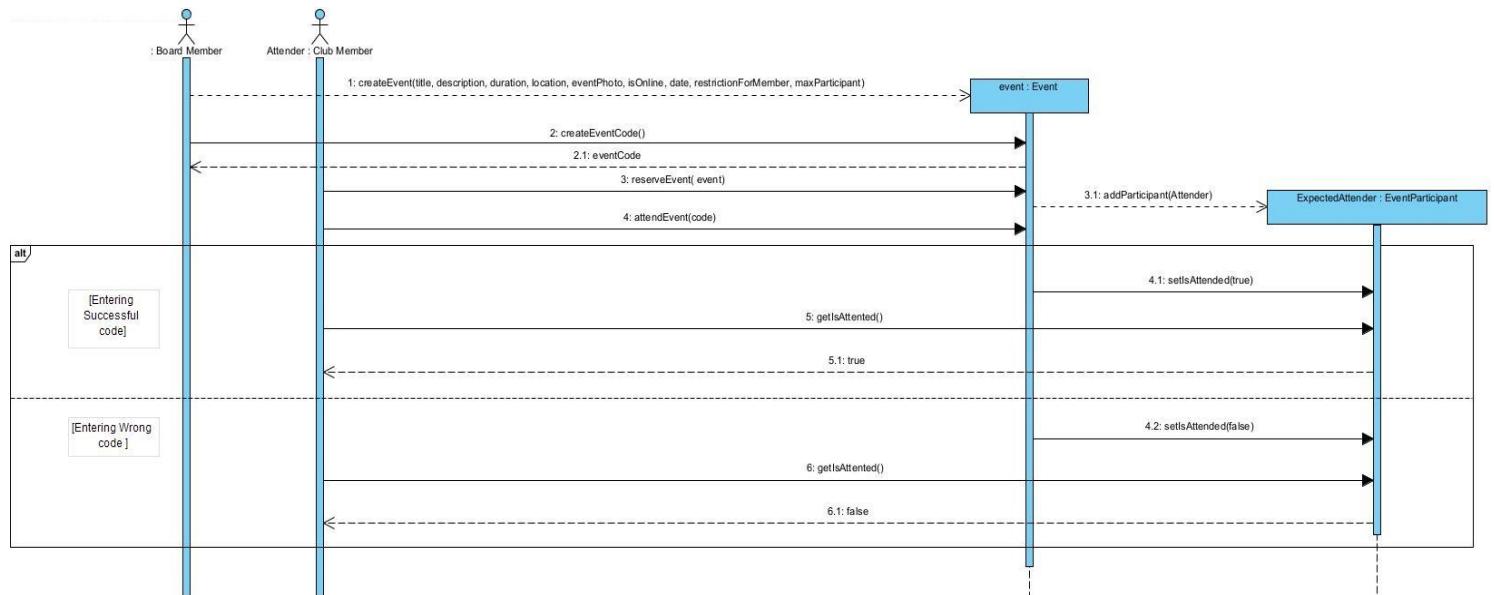


Fig. 3. Sequence Diagram - Lifecycle of Event Object.

An **event** object will be created after the **createEvent** function is called by the board member. This function will take details of the event as parameters such as title of the event, duration of the event, date of the event, etc. After the Event object is created, club board members will have the chance to create an attendance code by calling **createEventCode** function. When this function is called, it will return the event code to the board member. Before the start of the event, club members who are planning to attend the event, must reserve to that event with calling **reserveEvent** function. As a reply, this function will create an **EventParticipant** object which basically symbolizes that the owner of the object is expected during the event. During the event, board members will share the attendance code with the audience. In this part, attenders must specify that code to the software with calling **attendEvent** function. If the given code is true, **isAttended** attribute of the **EventParticipant** object will be set as true. Otherwise, it will be set to false.

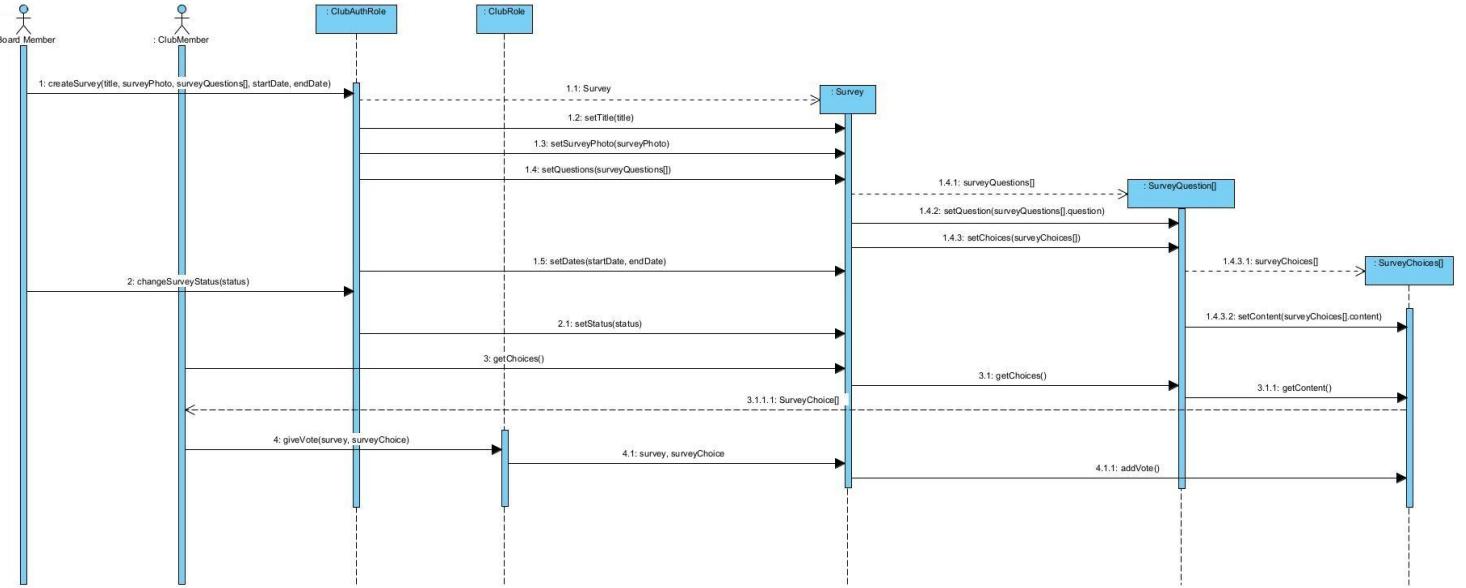


Fig. 4. Sequence Diagram - Life Cycle of the Survey Object.

A **Survey** object will be created by a **BoardMember** with calling **createSurvey** function. This function will take the details of the survey as parameters such as choices of the survey, start date of the survey, etc. Since the **createSurvey** function belongs to the **ClubAuthRole** object, this object will be called following methods to set properties of the survey object, **setTitle**, **setSurveyPhoto**, **setQuestions**, **setDates**. Also for each question and choice, a new **Question** and **Choice** object will be created and content of the choices and the question will be set using **setContent** and **setQuestion** functions. When the board member would like to share this survey with the club members, he/she will call the **setStatus** function. When a user would like to participate in this survey he will receive the choices with **getChoice** function and he will be able to vote to the survey with calling **giveVote** function which will call the **addVote** function of the related SurveyChoice object.

2.5.4.2 State Diagrams

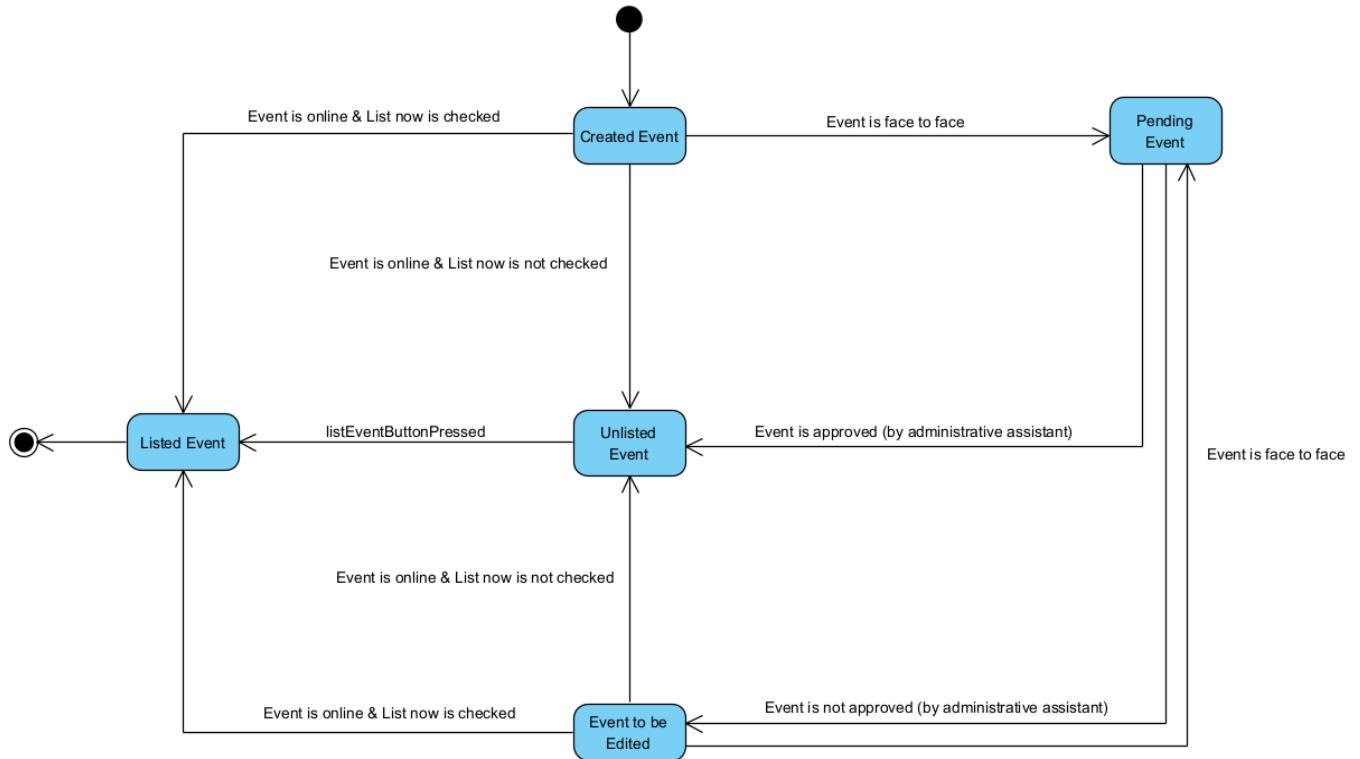


Fig. 5. State Diagram - Event Situations.

This state diagram shows the situations of an event. There are several options for an event. If the created event is face to face, an approval for time and place is needed, and the event becomes a pending event. If the event is approved by administrative assistants, the event becomes an unlisted event and ready to be listed. If the event time and place is not approved, the event becomes an event to be edited, and board members should edit the event time and place. After the edition is complete, the event goes to approval again. If the event is selected to be online, there are two choices. Board members can list the event immediately, or list it later. If board members choose not to list the event now, the event becomes an unlisted event. Unlisted events can be listed by selecting the “list” option.

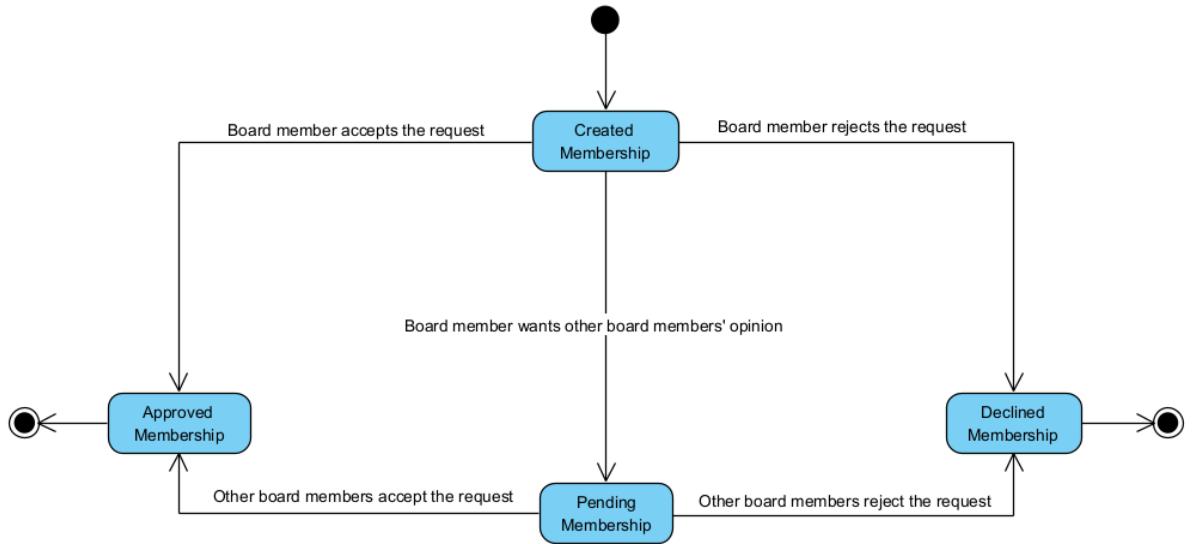


Fig. 6. State Diagram - Membership Situations.

This state diagram shows the situations of a membership to a club. When a membership request to a club is created by a student, there are several states. One of board members can accept or decline the request after seeing the request. There is one more option. If a board member wants the other board members' opinion about the request, he/she sends the request to the waiting list and the membership becomes pending membership. Other board members can accept or reject membership from the waiting list.

2.5.4.3 Activity Diagrams

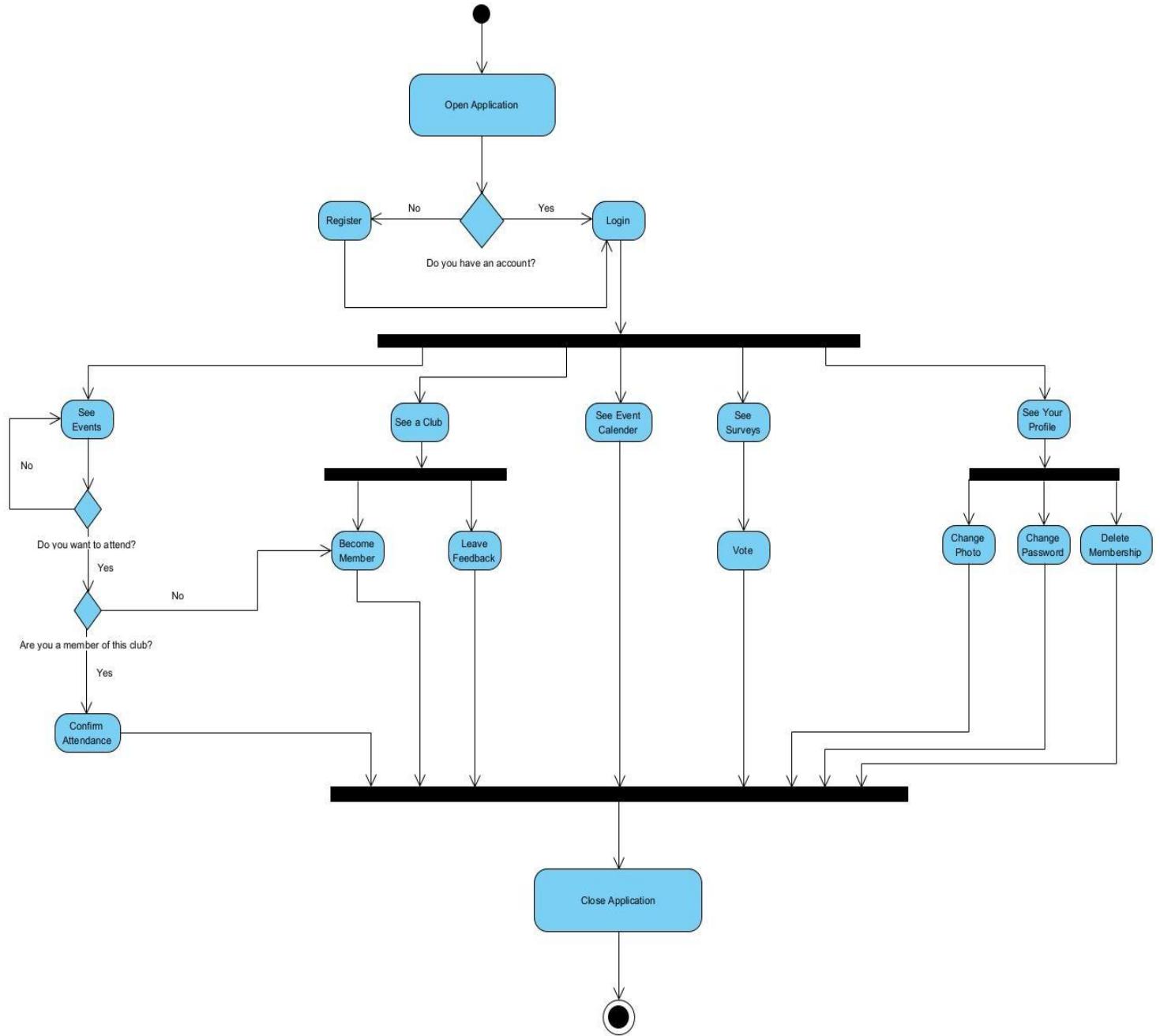


Fig. 7. Activity Diagram - Opening Application Case.

When a student opens the application, a login screen welcomes the user. If the student has an application account, he/she can immediately sign in to the account. However, if he/she does not have an account, he/she should go to the register page. After signing up to the website, he/she will again go to the login page.

On the main page, events of the clubs are displayed. Addition to that, the student can search for a club, see his/her event calendar, see existing surveys and see his/her profile. If the upcoming event attracts his/her attention, he/she can see the details of the event and have his/her name on the participants list by confirming his/her attendance. If there is a restriction by the club that only the members of the club can attend the event, then the student should become a member of the club first.

In the club details page, a student can become a member or a member of a club can leave feedback. In the event calendar page, there is a list of events that a student will attend and the past events that a student has already attended. In the surveys page, a student can participate in the club surveys that he/she is a member of. In the profile page, a student can change his/her profile photo, password and he/she can delete his/her club memberships.

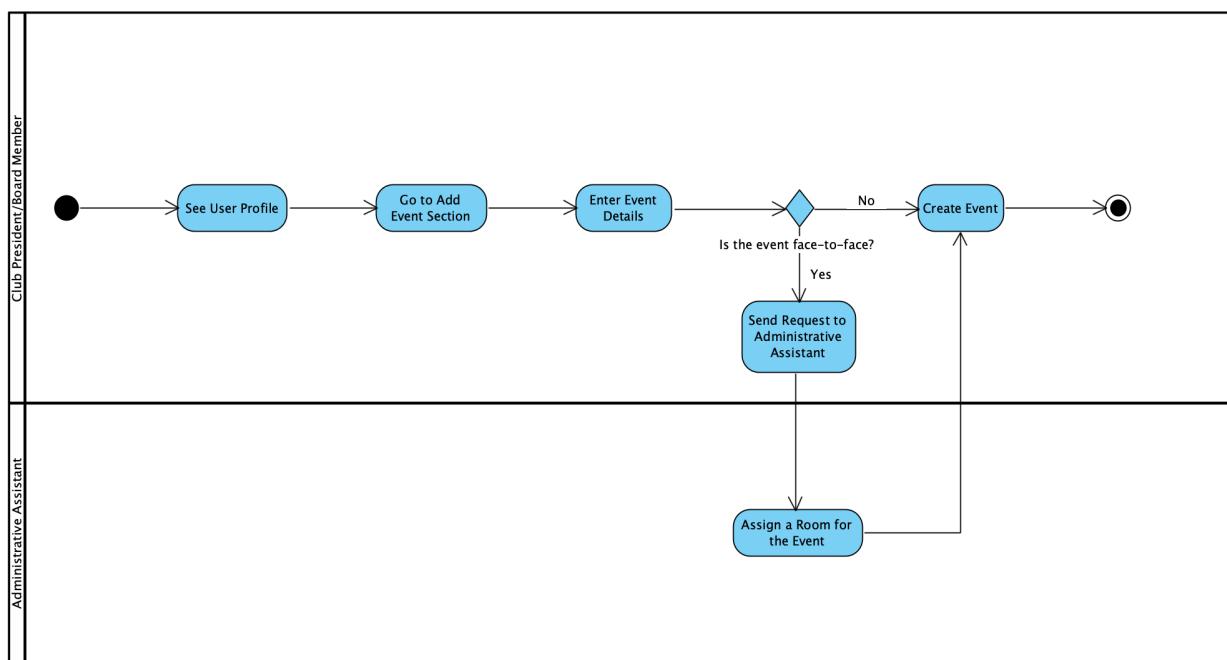


Fig. 8. Activity Diagram - Opening Event Page From User Profile Case.

In order to create an event, a board member or the club president should go to the club profile.

In the add event section, he/she should enter event details as well as the event's face-to-face or online information.

If the event will be held online, he/she can immediately announce the event. However, if the event will be face-to-face, firstly, the board member or the club

president have to send a request to the administrative assistant for a suitable room for the event.

After the administrative assistant reserves a room for the event, the club can announce the event for the students and the event creation process ends.

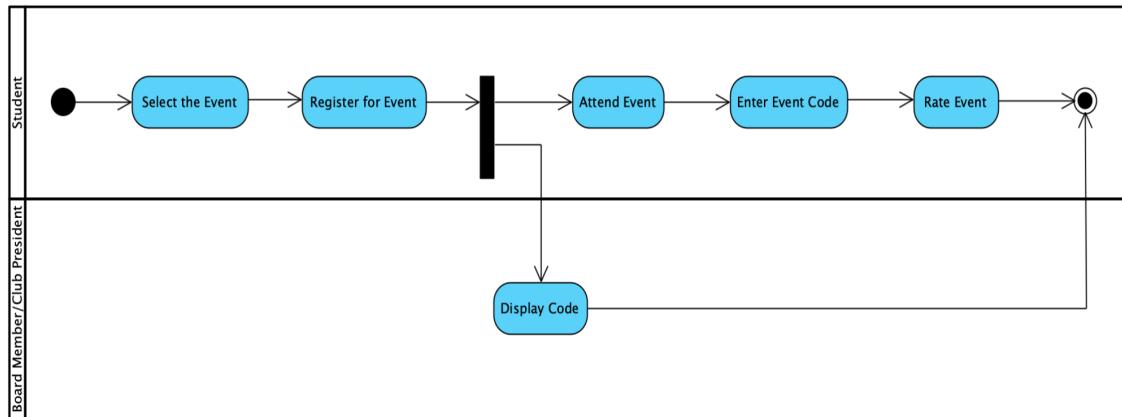


Fig. 9. Activity Diagram - Attending an Event Case.

When the event time comes, either the students or the board members should select the event.

In this page, the board members or the club president should create and display the code for the students to take attendance.

The students that attended the event should enter this code from this page to confirm their attendance. After the event, students can rate the event from this page again.

2.5.5 User Interface

About Bilboard Page

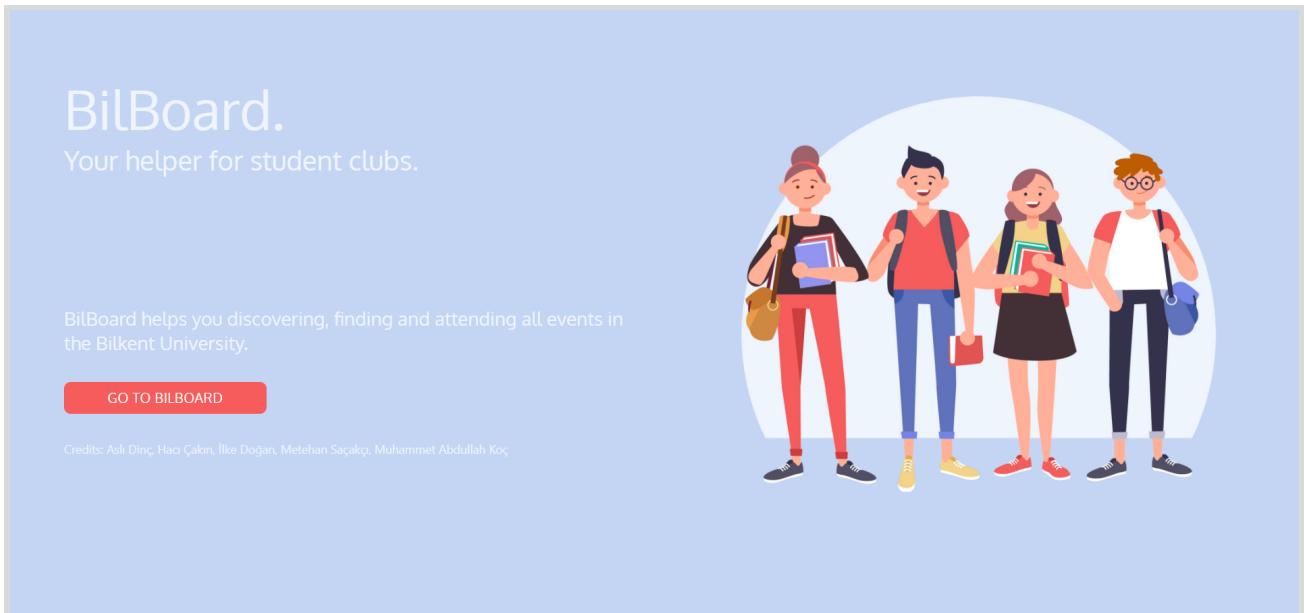


Fig. 10. About Bilboard Page.

The About Billboard page is used to redirect to the login and register page.

Login and Register Page

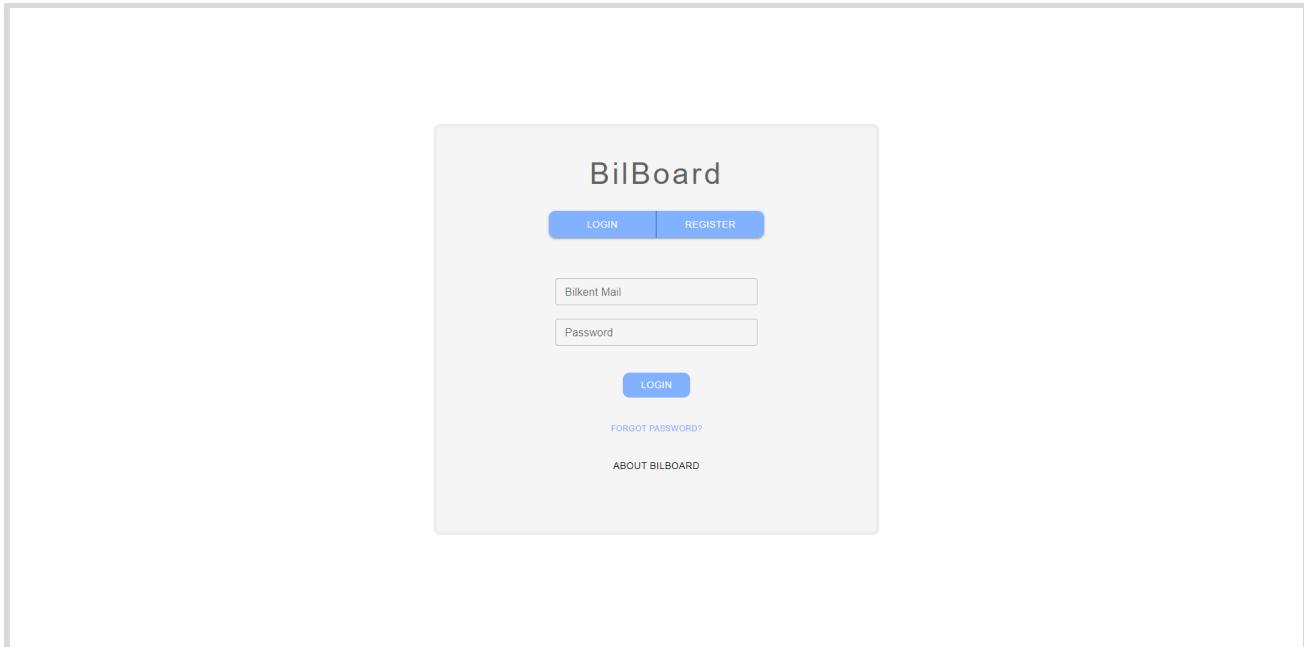


Fig. 11. Login and Register Page.

Main Page

A screenshot of the BiLBoard main page. The top navigation bar is blue with the BiLBoard logo on the left and a "ATTEND AN EVENT" button on the right. Below the navigation bar, there are two sections: "Events of Following Clubs" and "Discover". Both sections display cards for various events. In the "Events of Following Clubs" section, cards include: "Meltem Metin - StartMe" (Online, 250 participants), "FFB-06 P&G Marketing Days" (Online, 60 participants), "Career 101" (Online, 250 participants), "Peak Meetup" (Online, 250 participants), "Deloitte" (Online, 250 participants), and "Derin Uzayda Bilim" (Online, 100 participants). In the "Discover" section, cards include: "Art Arda Show" (Radio Bilkent, No limit participants), "FFB-06 P&G Marketing Days" (Online, 60 participants), "Career 101" (Online, 250 participants), "Peak Meetup" (Online, 250 participants), "Deloitte" (Online, 250 participants), and "Derin Uzayda Bilim" (Online, 100 participants).

Fig. 12. Main Page.

Students can attend an event by clicking the “Attend an Event” button on the top. “See All” buttons in this screen provide all events of following clubs, or events of clubs which the student is not a member of that club, in another

screen. Also, the search bar will always be in the navbar. Students can search events and clubs using the search bar.

Calendar Page

The screenshot shows the BilBoard calendar page. At the top, there's a blue header bar with the BilBoard logo on the left and a green 'ATTEND AN EVENT' button on the right. Below the header, there are two sections: 'Future Events' and 'Past Events'. Under 'Future Events', there are two event cards: one for 'Meltem Metin - StartMe' (Online, 250 participants) and another for 'OPERATIONAL RESEARCH KULÜBÜ GÜFRÜLA SUNAR!' (Online, 250 participants). Under 'Past Events', there are two event cards: one for 'Career 101' (IEEE Bilkent) and another for 'Peak Meetup' (IEEE Bilkent). The page has a light gray background and a clean, modern design.

Fig. 13. Calendar Page.

Hovering on an Event

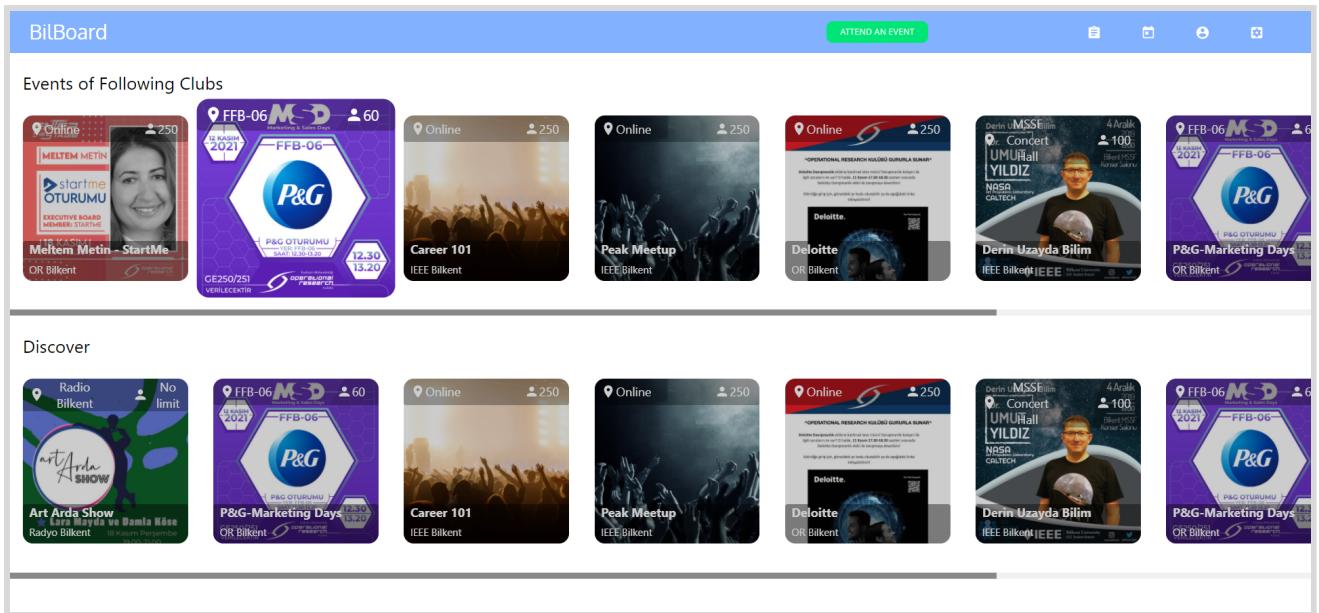


Fig. 14. Hovering on an Event.

If a user hovers on an event, event photo enlarges and event location becomes visible.

Event Page Seen by Students

The screenshot shows a detailed view of an event page. At the top, there's a blue header bar with the title 'BilBoard'. Below the header, the event details are listed: 'Event Name' with a star rating of ★★★☆☆, 'Club Name', and 'No.of'. To the right, there's a section titled 'Event Details' with a large amount of placeholder text. Below that is a section titled 'Event Location: Online'. On the left, there's a large image of two people sitting on grass, one writing in a notebook and the other looking at a laptop. Overlaid on the image is the text 'Language Club Activities for Universities and High Schools'. To the right, there's a section titled 'Question and Answer' containing four blue boxes, each with a question and answer pair. At the bottom right, there's a blue button labeled 'ASK A QUESTION'.

Fig. 15. Event Page Seen by Students.

When a user clicks on an event in one of the pages above, this event screen meets the user. The user can ask questions about the event and register the event. If the user has already registered for that event, he/she can unregister.

Club Page from Student User's Perspective

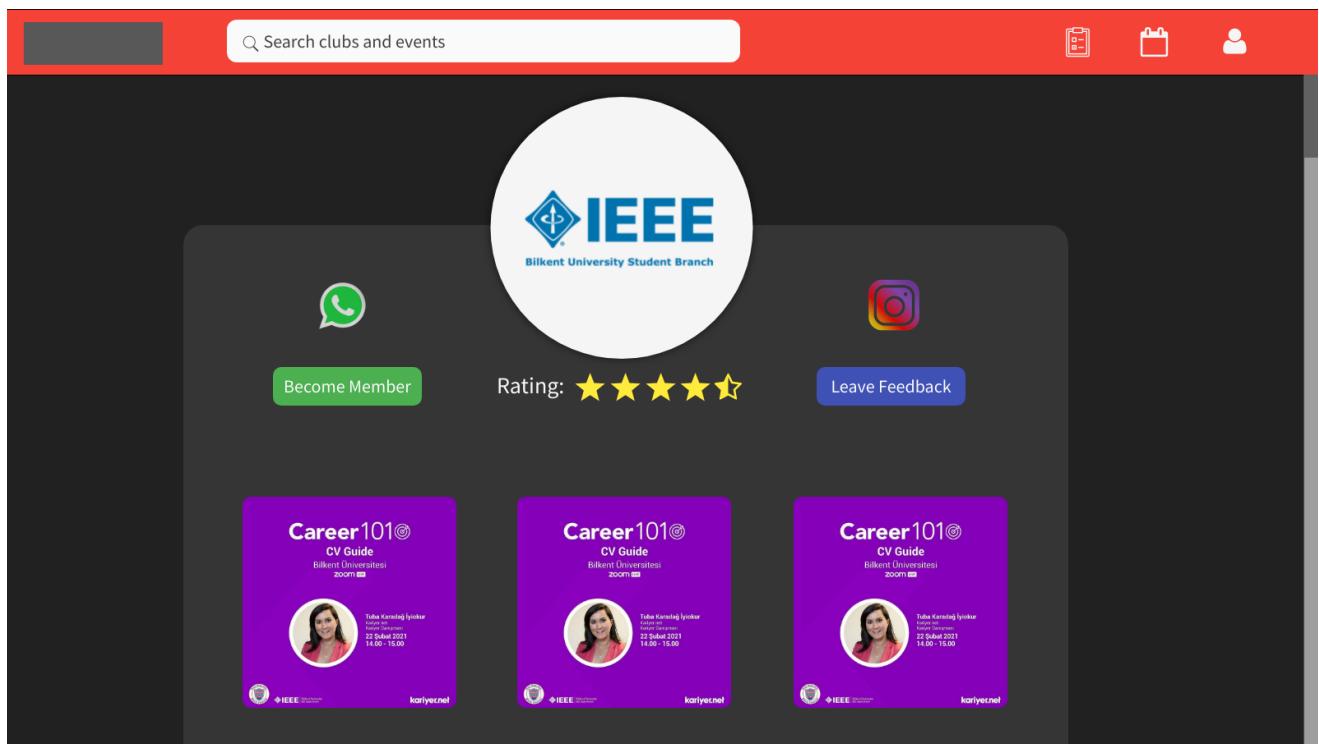


Fig. 16. Club Page from Student User's Perspective.

There are several functionalities of this page. If a user clicks WhatsApp or Instagram logo, he/she will be directed into the WhatsApp group of the club or Instagram page of the club. In addition, the average event rating can be seen on this screen. Users can also reach events by using this screen.

Leave Feedback Dialog

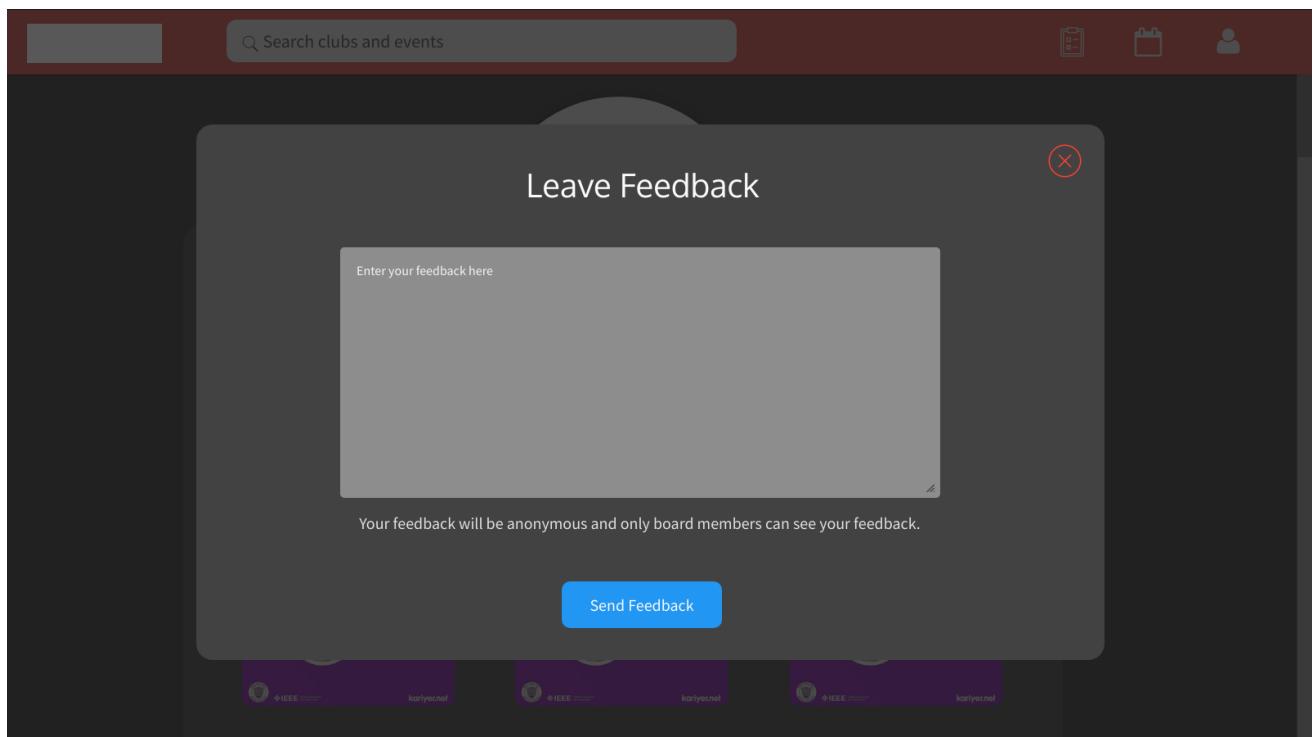


Fig. 17. Leave Feedback Dialog.

Surveys Page

The screenshot shows a list of six survey cards, each featuring the IEEE logo and the question "Question: What will our next event be?". Each card has a blue "VOTE" button on the right side.

Fig. 18. Surveys Page.

Attend Event Dialog

The screenshot shows a modal dialog titled "Attend An Event" with the instruction "Please enter 8-digit event code" and a text input field labeled "Code". Below the input field is a "SUBMIT" button and a "CANCEL" button. In the background, there are several event cards and a "Discover" section.

Fig. 19. Attend Event Dialog

A student can attend an event and become “attended” in the system. Also, GE250/251 points are not given if the student does not attend with code. If the code is wrong, the system warns the student. This screen appears when a student clicks the “Attend an Event” button on the navbar.

User Profile Settings Page

The screenshot shows the 'User Profile Settings Page' on a web application named 'BilBoard'. The top navigation bar includes a search bar, user profile icon, and links for 'ATTEND AN EVENT', 'LOG IN', and 'SIGN UP'. On the left, there's a circular profile picture placeholder and a 'Change Password' section with fields for 'Current Password', 'New Password', and 'Retype New Password', along with checkboxes for 'GE250/1' and buttons for 'UPDATE' and 'SIGN OUT'. On the right, under the heading 'Followed Clubs', there are four entries, each showing the IEEE logo and the word 'IEEE' in blue. Each entry has a 'GO TO CLUB PAGE' button (blue) and a 'LEAVE MEMBERSHIP' button (red).

Fig. 20. User Profile Settings Page

A user can change his/her password, set GE250/251 status, and manage memberships.

Club General Settings Page

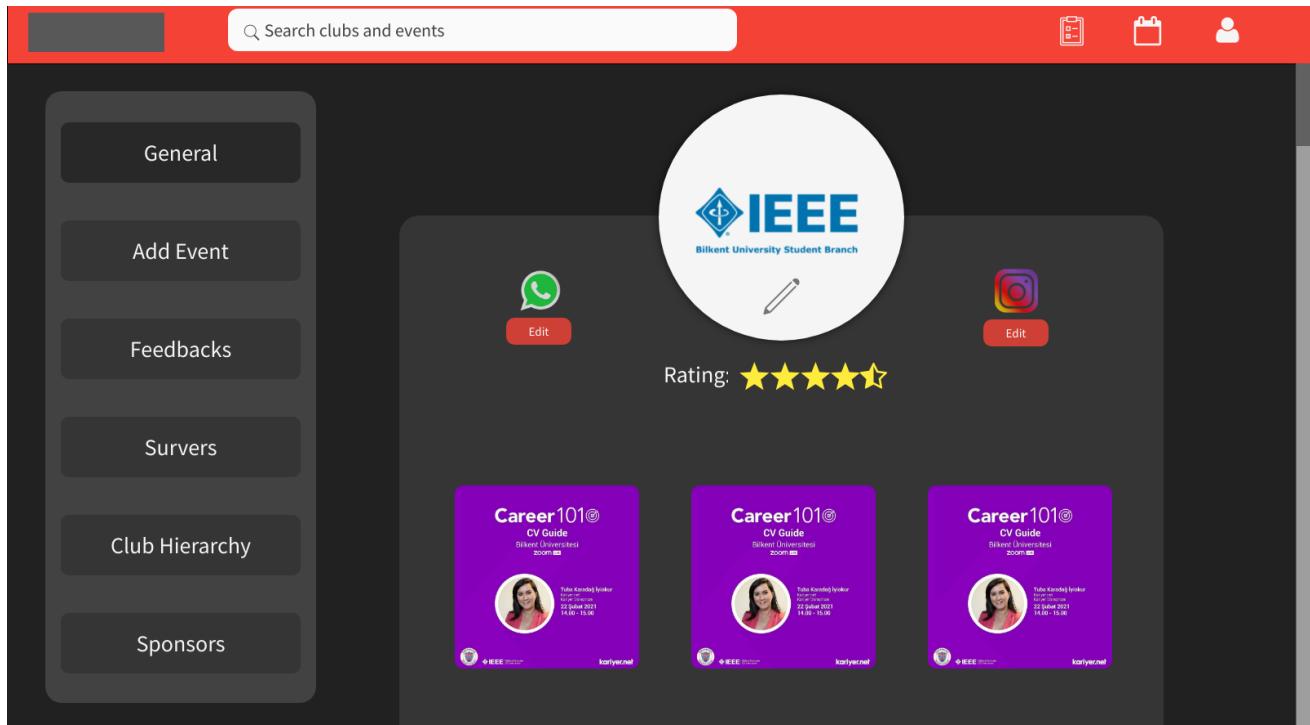


Fig. 21. Club General Settings Page.

General settings can be done in this screen (Change profile picture, edit WhatsApp and Instagram link etc.)

Add Event Page

The screenshot shows the 'Add Event' page within the BilBoard application. The left sidebar has a blue header 'BilBoard' and a menu with options: GENERAL, ADD EVENTS (highlighted in blue), FEEDBACKS, SURVEYS, CLUB HIERARCHY, and SPONSORS. The main area is titled 'Add Event'. It contains fields for Title (text input), Image (file upload icon), Description (text area), Date (date input), Location (radio buttons for Face to Face or Online, with Online selected), Enter Zoom Link (text input), List Now (checkbox), Number of Participant (text input), Ge 250/I Point (text input), and Private for Members (radio button). A blue 'ADD' button is at the bottom.

Fig. 22. Add Event Page.

A board member should give all information about the event that will be added. If one of the specifications (photo, title, description, date&time, location, number of participants) is missing, the board member cannot add the event. If the event is online, the event can be listed automatically.

Feedbacks Page

The screenshot shows the 'Feedbacks' section of the BilBoard application. At the top, there's a navigation bar with icons for file operations (New, Open, Save, Print, Find, Copy, Paste) and a 'BilBoard' logo. On the left, a sidebar menu includes 'GENERAL', 'ADD EVENTS', 'FEEDBACKS' (which is highlighted in blue), 'SURVEYS', 'CLUB HIERARCHY', and 'SPONSORS'. The main content area is titled 'Feedbacks' and displays three feedback entries, each in a light blue box:

- A feedback was given** at 20 October 2021. Includes 'ACCEPT' and 'DELETE' buttons.
- A feedback was given** at 20 October 2021. Includes 'ACCEPT' and 'DELETE' buttons.
- A feedback was given** at 20 October 2021. Includes 'ACCEPT' and 'DELETE' buttons.

Each feedback entry contains placeholder text about 'Lorem Ipsum'.

Fig. 23. Feedbacks Page.

Feedbacks were designed for general comments about the club. In order to provide privacy, all feedbacks are anonymous. However, feedback time is visible because a problem may be fixed after the feedback given.

Survey Settings Page

The screenshot shows the 'Survey Settings Page' for 'BilBoard'. The left sidebar includes links for GENERAL, ADD EVENTS, FEEDBACKS, SURVEYS (which is selected), CLUB HIERARCHY, and SPONSORS. The main area is titled 'Active Surveys' and displays five survey cards. Each card has a question, due date, vote count, and two buttons: 'CANCEL' and 'RESULTS'. A large blue 'ADD SURVEY' button is located at the bottom center.

Question	Due	Votes	Action Buttons
What will our next event be?	28 Oct 2021	25	CANCEL RESULTS
What will our next event be?	28 Oct 2021	1125	CANCEL RESULTS
What will our next event be?	28 Oct 2021	125	CANCEL RESULTS
What will our next event be?	28 Oct 2021	225	CANCEL RESULTS
What will our next event be?	28 Oct 2021	0	CANCEL

Fig. 24. Survey Settings Page.

All surveys are managed on this page. Current results of active surveys can be seen, or they can be ended. New surveys also can be added in this screen. All information should be provided in order to create a survey.

Club Hierarchy Page

The screenshot shows a mobile application interface for managing club hierarchy. At the top, there is a red header bar with a search bar containing the placeholder "Search clubs and events". On the right side of the header are three icons: a clipboard, a calendar, and a user profile.

The main content area has a dark background. On the left, there is a vertical sidebar with rounded corners containing several buttons:

- General
- Add Event
- Feedbacks
- Surveys
- Club Hierarchy
- Sponsors

In the center, there is a logo for "Bilkent University Student Branch IEEE" with the text "Bilkent University Student Branch" below it. To the right of the logo, the word "President" is displayed in red, followed by "Example Person". Below this, the text "Board Members" is also in red. A list of four names follows: "Example Person1" with a trash icon, "Example Person2" with a trash icon, "Example Person3" with a trash icon, and "Example Person4" with a trash icon.

At the bottom, there is a section titled "Add Board Member" in red. It contains two input fields: "Name" and "Bilkent ID", both with placeholder text. Below these fields is a red "Add" button.

Fig. 25. Club Hierarchy Page.

Board member information is provided here by the club president. If a president makes someone a board member, this member can have access to club settings.

Club Sponsor(s) Page

The screenshot shows the 'Sponsors' section of the BilBoard application. On the left, a sidebar menu includes options like GENERAL, ADD EVENTS, FEEDBACKS, SURVEYS, CLUB HIERARCHY, and SPONSORS, with 'SPONSORS' being the active tab. The main area is titled 'Sponsors' and lists five entries:

Logo	Name	Amount	Type	Action
	Garanti	10.000TL		
	Google	12.000TL		
	Microsoft	500.000TL		
	Cyberpark	1.000TL		
	Mobiversite	100.000TL		

At the bottom, there is a form with fields for 'Name', 'Amount', 'Type', and a blue 'ADD' button.

Fig. 26. Club Sponsors Page.

Club sponsors are important for clubs because clubs can organize events with the help of them. There are four types of sponsors: Platinum, Gold, Silver, and Bronze. Sponsors are classified according to their type of sponsorship.

Administrative Assistant Page

The screenshot shows a mobile application interface for managing room requests. At the top, there is a red header bar with a user profile icon on the right. Below the header, the title "Room Requests" is displayed. The main content area contains four separate request cards, each representing an event from "kariyer.net example event". Each card includes the IEEE logo, the event name, the number of participants (40), the preferred room (BZ-02), and a blue "Assign Room" button.

Event	No. of participants	Preferred room	Action
kariyer.net example event	40	BZ-02	Assign Room
kariyer.net example event	40	BZ-02	Assign Room
kariyer.net example event	40	BZ-02	Assign Room
kariyer.net example event	40	BZ-02	Assign Room

Fig. 27. Administrative Assistant Page.

This page is designed for administrative assistants. An administrative assistant can assign room for events. If the preferred room is full at the specified time, the assistant sends the request back and waits for a new request with a different room.

Event Page Seen by Board Members

The screenshot shows a mobile application interface for managing events. At the top, there is a red header bar with a search bar containing the placeholder "Search clubs and events". On the right side of the header are three icons: a clipboard, a calendar, and a user profile.

The main content area has a dark background. On the left, there is a purple rectangular card for an event titled "Career101@CV Guide" at "Bilkent Üniversitesi zoom LIVE". The card features a circular profile picture of a woman, Tuba Karadağ İyioğur, with her name and title below it. It also includes the date "22 Şubat 2021" and time "14.00 - 15.00". Logos for IEEE and Bilkent University are at the bottom of the card.

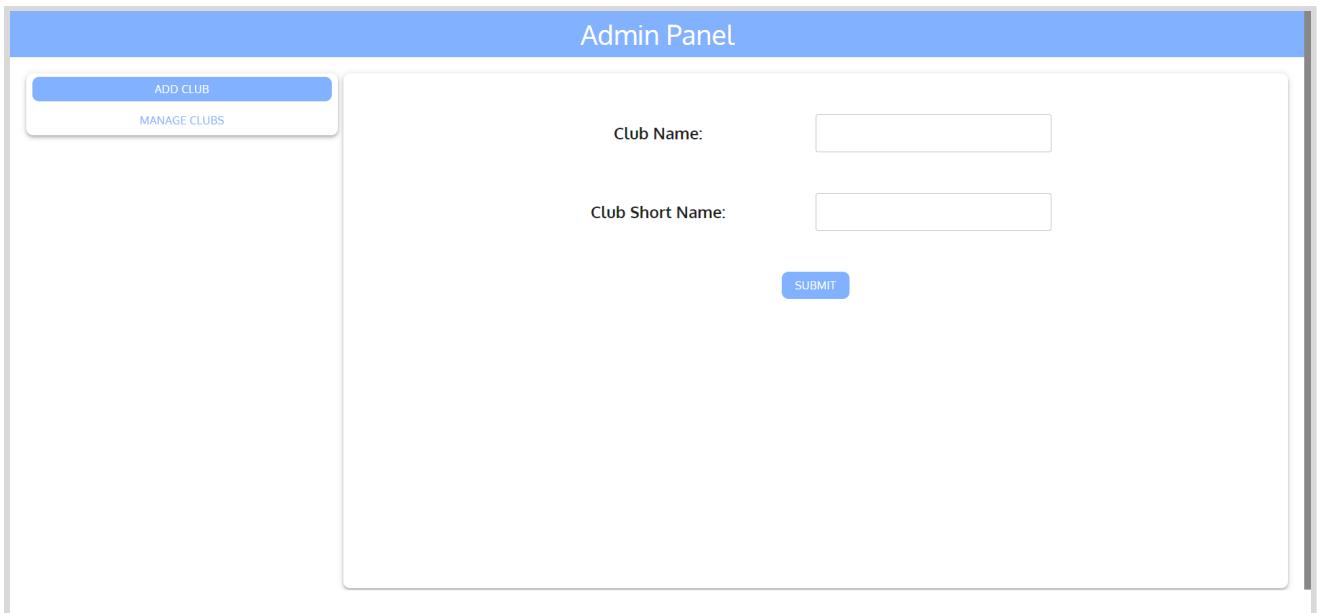
To the right of the event card, there is a section titled "Description of the event" with placeholder text. Below this is a green button labeled "Generate Code". Further down is a red button labeled "Edit Event".

At the bottom right, there is a section titled "Event Location" which is currently empty. To the right of this, there is a large gray box titled "Questions" containing three identical question cards, each asking "Q: Will GE250/251 points be given ?" followed by a blue "Answer" button.

Fig. 28. Event Page Seen by Board Members

This page is the core of an event. A board member can generate attendance code if the event time has come. Also, question answering is done on this screen. Moreover, if a board member clicks on the “Edit Event” button, the information of the event can be edited.

Admin Add Club Screen



The Admin Add Club Screen is a web-based form for adding a new student club. It features a blue header bar with the text "Admin Panel". On the left side, there is a vertical sidebar with two buttons: "ADD CLUB" (highlighted in blue) and "MANAGE CLUBS". The main content area contains fields for "Club Name" and "Club Short Name", each with an associated input box. A "SUBMIT" button is located at the bottom right of the form.

Fig. 29. Admin Add Club Screen

Admins can add a student club with this page.

Admin Manage Club Screen



The Admin Manage Club Screen is a web-based interface for managing existing clubs. It has a blue header bar with the text "Admin Panel". On the left, a sidebar contains the "MANAGE CLUBS" button (highlighted in blue). The central part of the screen displays a list of 13 clubs, each represented by a blue rectangular button labeled "CLUB 1" through "CLUB 13". To the right of the club list, there are fields for "Club Advisor" (containing "Advisor 5") and "Club President" (containing "President 5"), each with an associated input box. A "SAVE" button is positioned below these fields.

Fig. 30. Admin Manage Club Screen

Admins can manage a student club with this page (Setting club advisor and club president).

3 Glossary

AWS(Amazon Web Service): Amazon Web Services (AWS) is quite comprehensive. It is almost the most widely used cloud platform in the world (more than 200 services in data centers around the world). Developing companies, especially those who want to achieve this development fast, use the AWS cloud platform for better production by reducing costs with agile software [2].

EBS(Elastic Beanstalk): AWS Elastic Beanstalk makes it easy for software developers to take control of their applications on the AWS cloud platform. All that is required in this innovation is for the developer to install the desired applications. Elastic Beanstalk handles all the details of the deployment, such as controlling the application, snapshots, scaling, etc [3][4].

RDS(Relational Database System): Amazon Relational Database Service (Amazon RDS) is a service that enables and makes available the database setup and operation on the cloud platform. This database management saves both cost and time by providing self-scaling capacity. At the same time, this database service provides the ability to use the features of MySQL, MariaDB, Oracle, SQL Server or PostgreSQL databases. One of its most important features is that it automatically backs up the database created by the user and keeps it up-to-date.

JWT(JSON Web Token): JWT(JSON Web Tokens), JSON-based RFC 7519 standard, is an open standard that provides data exchange between the parties using the application. It enables the creation and sending of the requested data between the server and the client. Widely used in the world, JWT allows users to protect user rights such as web service security, cyber security, and authentication [5].

CORS(Cross-Origin Resource Sharing): In the context of web browsers, a site can normally only allow XMLHttpRequest* (also called ajax) addresses in their domain. Some sites may want to allow requests from other domains. This is called Cross-Origin Resource Sharing (CORS). This is done with the access-control-allow-origin header. A web page can freely add cross-root images, style sheets, and videos [6].

IMAP(Internet Message Access Protocol): With the increasing use of the internet, the need to send emails over the internet has emerged due to the difficulty of accessing the internet and the need for speed. To meet this need, IMAP is an email protocol for exchanging mail over the internet and provides an application layer that allows local users to access a remote email server [7].

4 Improvement Summary

This iteration of the analysis report has some improvements over the first iteration after feedback.

4.1 Introduction

Because the role of advisor has changed, improvements were done in introduction for advisors.

4.2 Proposed System - Functional Requirements

Some types of users and their abilities have changed. Non-member role is added, and the advisor's abilities were updated. Also, the roles were arranged in a hierarchical order.

Rather than focusing on the page structures, we focused on the functionality of our application in this section. All of the functional requirements were updated accordingly. In addition, three functions (Arrange Building/Class Condition, Viewing Club Members, Assigning Specific Roles) were added.

4.3 Non-functional Requirements - Reliability

After the feedback, we removed the image size limitation part in the Reliability section.

4.4 Non-functional Requirements - Scenarios

Because of the change in Advisor role, existing scenarios were updated.

4.5 Use Case Diagram

The diagram was bounded. Advisor became a part of the hierarchy and ability of viewing club members was given to the Advisor. Names of the use cases were edited. New use cases were added to administrative assistants (Add/remove building and classes), and admins (Create/delete a club, assign a club president and advisor). Also, some use case diagrams that have been in iteration 1 were removed.

4.6 Class Diagram

Associations were updated according to the feedback given. The SurveyQuestion class was added and bound to SurveyChoices. EventPoint class was removed. The point field is connected to the EventParticipant class. The issue about administrative assistants was fixed.

4.7 Sequence Diagram

Lifecycle of the Event object and Survey object is created. Existing diagrams were removed. Explanations were edited.

4.8 State Diagram

State diagrams of an event and membership were created. Existing diagrams were removed and explanations were edited.

4.9 Activity Diagram

Login action was deleted for the Create Event and Attend Event case. In the Attend Event case, all diagram were included in swimlanes.

4.10 User Interface

Screenshot of screens which were implemented before were added and their UI templates were removed. Because some of the screens have not been implemented yet, their templates remain. Also, admin screens were added.

5 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.
- [2] K. Yadav, "What is AWS and What can you do with it". [Online]. Available: <https://medium.com/@kunalyadav/what-is-aws-and-what-can-you-do-with-it-395b585b03c>. [Accessed: 29 October 2021].
- [3] "AWS Elastic Beanstalk". [Online]. Available: https://aws.amazon.com/elasticbeanstalk/?nc1=h_ls. [Accessed: 29 October 2021].
- [4] B. Lutkevich , "Amazon RDS (Relational Database Service)". [Online]. Available: <https://searchaws.techtarget.com/definition/Amazon-Relational-Database-Service-RDS>. [Accessed: 29 October 2021].
- [5] "Introduction to JSON Web Tokens". [Online]. Available: <https://jwt.io/introduction>. [Accessed: 29 October 2021].
- [6] "Cross-Origin Resource Sharing (CORS)". [Online]. Available: <https://developer.mozilla.org/en-US/docs/Web/HTTP/CORS>. [Accessed: 29 October 2021].
- [7] "Internet Message Access Protocol (IMAP)". [Online]. Available: <https://www.gartner.com/en/information-technology/glossary/imap-internet-message-access-protocol>. [Accessed: 29 October 2021].