



2020-2021 SPRING SEMESTER

CS319 - OBJECT ORIENTED SOFTWARE ENGINEERING ITERATION 1 PEEREVIEW ANALYSIS REPORT

GROUP 1B MEMBERS

ABDUL RAZAK DAHER KHATIB 21801340

MUHAMMAD SALMAN AKHTAR SOOMRO 21701446

EGE MOROĞLU 21401240

UTKU GÖKÇEN 21703746

YİĞİT DİNÇ 21704275

1 Introduction	6
2 Overview	6
2.1 Users	6
2.1.1 Instructor	6
2.1.2 Grader	6
2.1.3 Student	7
2.2 System Components	7
2.2.1 Course	7
2.2.2 Project	7
2.2.3 Group	8
2.2.4 Assignment	8
2.2.5 Evaluation	8
2.2.6 Review	9
2.2.7 Response	9
2.2.8 Graphs	9
3 Functional Requirements	9
3.1 Create a course	9
3.2 Create a Project	9
3.3 Create Groups	9
3.4 Find a Group	9
3.5 Submissions	10
3.6 Evaluations	10
3.7 Grading	10
3.8 Add graders	10
3.9 Add and remove Students	10
3.10 Notifications	10
3.11 Responses	10
3.12 Mail Services	10
3.13 Integration with Slack Application	11
3.14 Analysis and Visualization of Evaluations	11
4 Non-Functional Requirements	11
4.1 User Friendly Interface	11
4.2 Privacy	11
4.3 Availability	11
4.4 Minimalist	12
4.5 Easy to Navigate	12
4.6 Integration with Other Services	12
4.7 Reliability	12
4.8 Performance	12

5 System Models	12
5.1 Use Case Model	12
5.1.1 Use Case Scenarios	13
5.1.1.1 Use Case: Join Group	13
5.1.1.2 Use Case: Evaluate Groupmates	14
5.1.1.3 Use Case: Evaluate Other Groups	14
5.1.1.4 Use Case: Create Group	15
5.1.1.5 Use Case: Manage Requests	16
5.1.1.6 Use Case: Invite Members	16
5.1.1.7 Use Case: Max number of groups reached	17
5.1.1.8 Use Case: Discuss Evaluations	17
5.1.1.9 Use Case: Resubmit	18
5.1.1.10 Use Case: Post a response	19
5.1.1.11 Use Case: Start Slack Discussion	19
5.1.1.12 Use Case: Create Course	20
5.1.1.13 Use Case: Create Project	21
5.1.1.14: Use Case: Assign Students to Groups	21
5.1.1.15 Use Case: Create Assignment	22
5.1.1.16 Use Case: Submit Assignment	23
5.1.1.17 Use Case: Grade Assignment	23
5.1.1.18 Use Case: Send Message to Another User	24
5.1.1.19 Use Case: Join Project	25
5.2 Object and Class Models	26
5.2.1 Package Diagram	26
5.2.2.1 User Student Grader Diagram	27
5.2.2.2 User Instructor Relationship	28
5.2.2.3 User Course Relationship	29
5.2.2.4 Course Project Relationship	30
5.2.2.5 Group Project Relationship Figure 5.2.2.5 Group Project Relationship	31
5.2.2.6 Comment Review Response Submission Relationship	32
5.2.2.7 Evaluation Review CompletedSubmission Submission Note Relationship	33
5.2.2.8 Review Assessment Relationship	34
5.2.2.9 Group Course Diagram	35
5.2.2.10 Student Group Diagram	36
5.2.2.11 Manager Class	37
5.2.2.12 Notification Class	38
5.2.2.13 Date Class	38
5.3 Sequence Diagram	39
5.3.1 Course Creation	39
5.3.2 Grader Grading Submission	40
5.3.3 Group Formation	41

5.4 Activity Diagram	43
5.4.1 Reviews	43
5.4.2 Group Creation	45
5.5 State Diagram	45
5.5.1 Submissions	46
5.5.2 Groupmates Evaluations	48
5.6 Screen Mockups	49
5.6.1 Login As Page	50
5.6.2 Sign Up Page	50
5.6.2.1 Sign Up Page For Students	51
5.6.2.2 Sign Up Page For Instructor and Grader	52
5.6.3 Login Page	53
5.6.4 Main Page	53
5.6.4.1 Main Page For Students	54
5.6.4.2 Main Page For Instructor and Grader	56
5.6.4.2.1 Drop Menu For Instructor and Grader	56
5.6.5 Profile Pages	56
5.6.5.1 Profile Page For User	57
5.6.5.2 Profile Page For Other User	58
5.6.6 Courses Pages	59
5.6.6.1 Courses Page For Students	59
5.6.6.2 Courses Page For Instructor	59
5.6.6.3 Courses Page For Grader	60
5.6.7 Detailed Course Pages	61
5.6.7.1 Detailed Course Page For Student	61
5.6.7.2 Detailed Course Page For Instructor	62
5.6.7.3 Detailed Course Page For Grader	63
5.6.8 Assignments Page	64
5.6.9 Detailed Assignment Pages	65
5.6.9.1 Detailed Not Completed Assignment Page For Students	65
5.6.9.1.1 Detailed Completed Assignment Page For Students	66
5.6.9.2 Detailed Assignment Page For Instructors	67
5.6.9.3 Detailed Assignment Page For Grader	68
5.6.10 Groups Pages	68
5.6.10.1 Groups Page For Students	69
5.6.10.2 Groups Page For Instructor And Graders	70
5.6.11 Groups Assignments Page For Students Instructor and Graders	71
5.6.12 Grading Page For Instructors and Graders	72
5.6.12.1 Grade Group Page For Instructors and Graders	72
5.6.12.2 Grade Individually Page For Instructor and Graders	73
5.6.13 Detailed Group Page	74
5.6.13.1 Detailed Your Group Page For Students	74
5.6.13.2 Detailed Other Groups Page For Students	75
5.6.13.3 Detailed Other Groups Page For Instructor and Grader	76

5.6.14 Evaluation Pages	77
5.6.14.1 Evaluate Your Group Member Page	77
5.6.14.2 Evaluate Other Groups Page	78
5.6.15 Create Course Page For Instructors	79
5.6.16 Create Project Page For Instructor	80
5.6.17 Assign Student to Groups Page	81
5.6.18 Grades Page	82
5.6.19 Notifications Page	82
5.6.20 Enroll in a Course Page For Students	83
5.6.21 Create Assignment Page For Instructors	83
5.6.22 Create Group Page	84
5.6.23 Find Group Page For Students	85
5.6.24 Send Request To Join a Group For Students	85
5.6.25 Change Password Page	86
5.6.26 Review Responses to the Reviews Page	87
6 Summary and Improvements	88

1 Introduction

Peer grading system is a web-based application where students can create their own project groups and grade their teammates as well as other groups' works. This system will not only provide a grading feature but also allow other actions which can be performed by instructors and graders. Instructors will assign homeworks and graders will be able to track the uploaded assignments and grade them.

The main goal of the application is evaluating teammates and other groups' members and commenting on their contribution on the project. Additionally, every group will post their works on their group's page and students from other groups will be able to see and evaluate their work. Users will be able to respond to others' reviews and discuss with them. Application will contain different courses which a student is supposed to take. Projects, groups and assignments will be specific for each course.

Project process will start with forming groups. Students will be able to find groups and send requests to join them or create their own group and find members for their groups. Once the groups are formed, each group will have their own group page where they will share information about their projects and upload their works. During the semester students will see other groups' works and at the end of the semester they will grade their teammates.

2 Overview

2.1 Users

2.1.1 Instructor

The instructor user is made for instructors or lecturers of the course. The user can create courses, where they can create projects, add assignments and grade them, assign graders, and evaluate students. Instructor users can also create a project where they can add students, add assignments, and most importantly give the students the chance to evaluate each other as group mates or other groups. This user will also have the authority to access and see most components of the system from students grades to evaluations to the chemistry of a group as well as respond and endorse responses and reviews. Instructors will also have the ability to limit users accessibility and visibility of evaluations and projects pages and also remove students.

In short the instructor users will have full control over the course and all components associated with it.

2.1.2 Grader

The grader user is a user assigned by Instructor to grade assignments. This user has limited authority and can only interact with students and in the allowed scope by Instructor. The user can view and grade assignments in a course or a project. This user

can also respond to reviews and endorse them, and view evaluations and grades if given the authority by the instructor.

2.1.3 Student

Student user is on the opposite end of the Instructor user type. This user can join courses created by the Instructor, submit assignments, if projects exist then Students can join projects, if Instructor lets, Students may create groups, may join groups formed by other Users, and might leave the group they are in. Main purpose of such engagement is to let Students have experience with other groups, hence to have deeper criteria for evaluating themselves as group members.

The main role of this user will be in a project where they can submit their assignments and review others' work. They can submit assignments, evaluate their group mates' contributions to each assignment according to some specific criteria and their overall contributions to the project. Students can also evaluate other groups after viewing their assignments. Their authority, however, can be limited by the instructor, so that they cannot see evaluations of others or their group's chemistry. The user info and her grades and evaluations will be visible on her page to the allowed users assigned by instructors.

2.2 System Components

2.2.1 Course

Course is the main component of the system with it holding all other components like projects and groups. The Instructor user type will be able to create this component and assign a unique code to it. Any Student user type will be able to join the course using that code. Course will have various components, like education materials, assignments, and projects.

In the education materials the instructor will be able to post any form of material related to the course. In assignment the instructor can post any assignment for the students. Instructors will also have the ability to create one or more projects and add students to it. In case the instructor needs help with grading, the user will be able to assign graders to the course for the assignments posted.

2.2.2 Project

Project is a quite broad term in Bilkent University, in one course Project can be a term project with five students requiring more than one submission, or a one week project with three students requiring one submission. To correctly provide different types of projects, Instructors have the opportunity to set their rules as they wish on the system. Outline of a project consists of its title, its description, its deadline, its final grade and student groups. Basically a project is an assignment that requires a group. Although gathering information about a project is complicated, every project has its own forum to ease the communication between students, graders, and instructors. Thus, while helping students to inform themselves, form groups may ease the burden on the instructors back by enabling them to control the environment that the information is given.

2.2.3 Group

At the very beginning of the semester, students form their project groups. They are able to form their groups by finding groups and sending joint requests or creating groups and then finding members. If there are many students who can not find groups, instructors may assign random groups. If there are a few students who can not find groups, instructors may assign them to random groups. Once students form their groups, they will create their group pages. There will be a link to groups github project and every group will be responsible for sharing their github links. They will also specify their projects' features in a README file. Students will be able to see groups' pages and their submissions as well as their ratings from evaluations.

2.2.4 Assignment

Assignments are given by the instructor and have specific deadlines. Assignments given by the instructors are graded and Graders and Instructors have permission to grade submitted Assignments. Students need to submit the specific assignment before the certain deadline. Grading policy and criteria depend on the instructor. Graders are able to grade submissions in the light of instructor's permission. Instructor gives the description and the deadline of the assignment when assigning the specific assignment. Students are required to drag and drop their submission files before the deadline.

2.2.5 Evaluation

Evaluation is one of the main components of the system. This component represents the evaluations given by Student users to each other, or to other groups. It also represents the evaluations given to students by graders and instructors. These components can also be used to represent evaluations given by students to their graders. Evaluations are available for each assignment for group mates to evaluate each other and for overall evaluations of group mates for the project as a whole. Students will be able to evaluate other groups after looking at their assignments, where this component is useful.

Evaluations will be represented in different ways. During the evaluation process a score out of ten will be assigned according to predetermined criteria and add notes, the average of the scores is calculated and saved. Then when a group or a student page is accessed the average evaluations given by others will be represented by a graph showing the change throughout the project. Evaluations will also be used to calculate Chemistry of a group, by taking the average of the averages of each group mates' evaluation to others. The Chemistry attribute will be used to show how satisfied the students are with their group mates' performance and can be helpful in informing the instructor how useful her way was in assigning groups. The importance of the Chemistry attribute can also be seen where the instructor can tell if the group had problems cooperating as a whole or whether there were some students with poor performance or communicating skills.

2.2.6 Review

Review is what the student leaves as a note after evaluating others' work, it can also be a note for graded submissions. They are important because they are, along with grading according to some criteria, the main way to evaluate others.

2.2.7 Response

As a complementary to the review component the user can respond to a review submitted by others and they can also respond to the response, building a platform for forum-like discussion under each submission.

2.2.8 Graphs

Using the data collected during the grading and the evaluation processes we will implement visualization tools to show the average of the grades and evaluations, their change after time, the outliers of each group, the average evaluation of each member, the average evaluation by other groups, and others. These graphs would show in the groups' page and in the assignment page. Their visibility could be limited based on how sensitive the data is, for example the average grade of a student should not be visible to anyone but the instructor and the student.

3 Functional Requirements

3.1 Create a course

The Instructors should be able to create their courses where they will create projects and add students and assignments and material.

3.2 Create a Project

The instructor should be able to create projects and assign graders and add students to it.

3.3 Create Groups

The students can create groups and add others to them, they can also submit offers and accept them. The instructor can also assign random groups.

3.4 Find a Group

The students will be able to look for groups and submit offers based on their skills. The students will be able to find groups with vacant places.

3.5 Submissions

Submissions are a key requirement in our system since they are the backbone of the evaluations. The users will have the ability to add submissions for assignments and edit them later for any second cycle of submissions or in discussions

3.6 Evaluations

The students will be able to evaluate their group mates after each assignment and at the end of the project. The evaluation process for multi-cycle assignments will be continuous and extensions of each other. They will also be able to evaluate other groups' submissions.

3.7 Grading

The instructors and the graders will be able to grade assignments of the students.

3.8 Add graders

For every course and project the instructor can assign graders who can grade all the assignments.

3.9 Add and remove Students

The instructor will be able to add students to projects and groups and remove them.

3.10 Notifications

The students will be notified when any important event happens like a creation of a new project or an assignment being posted.

3.11 Responses

The students will be able to respond to reviews made to their assignments, they will be able to respond to responses and so on, creating a discussion platform.

Additionally, they will be able to make edits to their submissions and include these responses to help better show the changes that they have made throughout the discussion process and discuss these changes.

3.12 Mail Services

With the assignments, grading, evaluations, and discussions all included, the need for mail services to be included is great. The service will be used to remind

students of deadlines for submissions and evaluations, as well as notify them of new responses and reviews.

3.13 Integration with Slack Application

In order to enhance the experience of the user and make detailed discussions possible the application will provide the option for users to take their discussion to a Slack chat room.

3.14 Analysis and Visualization of Evaluations

With all the evaluations grades being submitted to the system it would be a waste not to analyze them and provide a visualization showing how well did the group work together and what are the contributions of each member throughout the project, the potential for this data is great, from making it easier for instructor to grade each member to help students see the average of impression of their work on other groups' members.

4 Non-Functional Requirements

4.1 User Friendly Interface

The application provides a simple and user friendly interface that helps users to use the application efficiently. Users can easily access the peer evaluation pages which is the main purpose of the application. In addition, by help of the colors, icons and widgets used, an easy-to-use application is offered to the user so the users can easily locate important components and interact with them

4.2 Privacy

In some cases, students may not want their grades to be seen by others. For this reason, in the application, students will only be able to see their own grades. Only instructors and teaching assistants will see all students' grades. Additionally the evaluations will be anonymous. Thus, privacy will be provided for students.

4.3 Availability

The application will be accessible 24/7. In this way, students will be able to post assignments and check announcements and ask questions and communicate with each other whenever they need.

4.4 Minimalist

The interface is simple and minimalist to make it easy for the user to use and the machine to run.

4.5 Easy to Navigate

The application is very intuitive, and the user can reach any page through simple and logical sequence, so no confusion in how she can access a specific feature. Additionally, it will have shortcuts to reach important components of the system. Interface is developed keeping in mind familiarity with other such applications like Moodle, Slack etc. So the user can navigate around the application easily.

4.6 Integration with Other Services

Integration with other services is one of the main requirements since we cannot implement all features needed because of the limited time and resources.

4.7 Reliability

Since the users will use the services to submit their assignments the system must be reliable and work 24/7 to prevent any technical issues from delaying the submission process.

4.8 Performance

The application will run on online servers with minimal amount of work on clients' devices and therefore it will run very fast, less than one second for any operation, while for the uploading process it should not take more than 5 seconds for any file with a reasonable size (less than 100 MB).

5 System Models

5.1 Use Case Model

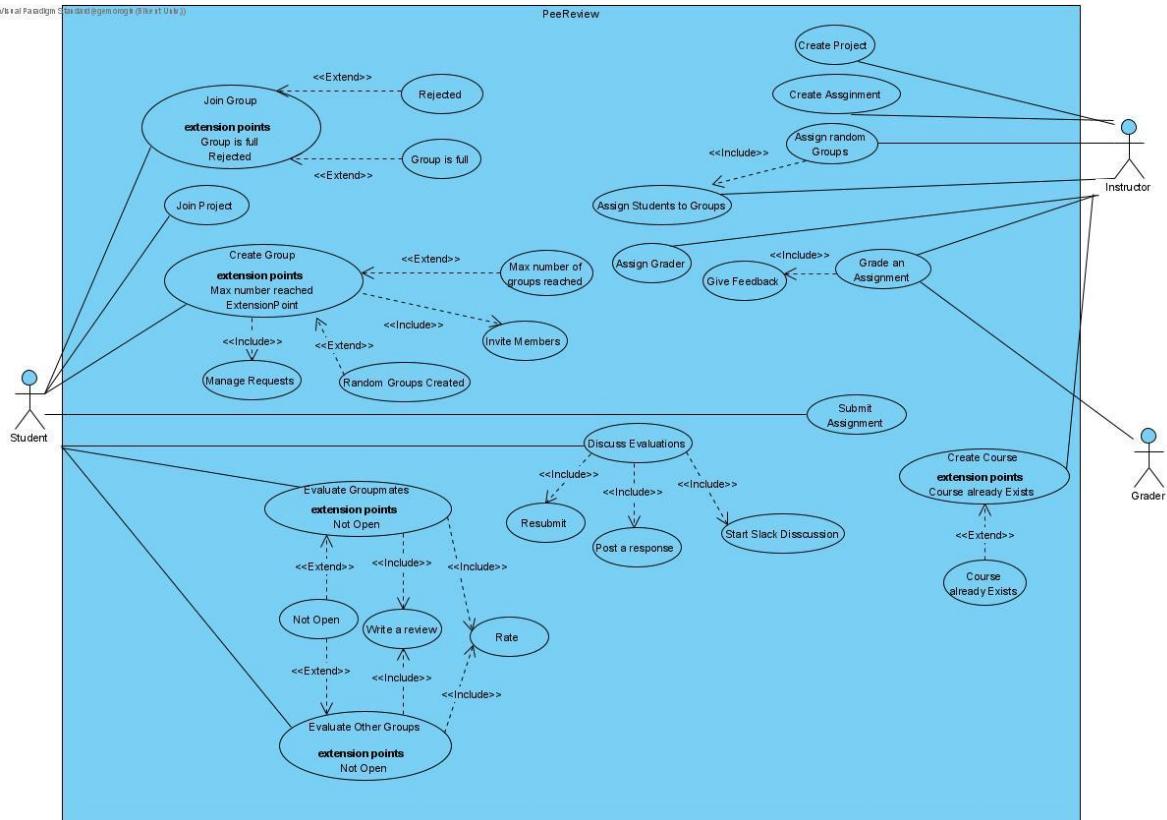


Figure 5.1 Shows the Use Case model of the Peer Review Application

5.1.1 Use Case Scenarios

5.1.1.1 Use Case: Join Group

Participating actors: Student

Pre-conditions:

- Student should not be in a group

Post-condition:

- Student is a member of a group

Entry condition:

- Student creates a group or finds an available group

Exit condition:

- Student goes to another page

Flow of events:

- Student creates a group and finds members
- If student is not in a group finds a group
- If group is available student request to be a member
- Group accepts the request
- Student joins to the group

Special/Quality requirements:

Joining a group may take a while depending on the group's response.

5.1.1.2 Use Case: Evaluate Groupmates

Participating actors: Student

Pre-conditions:

- Evaluation must be enabled

Post-condition:

- Student's total grade is effected

Entry condition:

- Student must be in the "Evaluate your group members" page

Exit condition:

- Student goes to another page

Flow of events:

- Time for evaluation is reached
- Student goes to "Evaluate your group members" page
- Student rates groupmates out of 10
- Student leaves a note about groupmates

Special/Quality requirements:

Evaluation may take a while. However, submit time should take less than 5 seconds

5.1.1.3 Use Case: Evaluate Other Groups

Participating actors: Student

Pre-conditions:

- Evaluation must be enabled

Post-condition:

- Group's total grade is effected

Entry condition:

- Student must be in "Evaluate Groups" page

Exit condition:

- Student goes to another page

Flow of events:

- Time for evaluation is reached
- Student goes to "Groups" page and selects a group
- Student checks the submission file of the group
- Student rates the group out of 10
- Student leaves on group's work

Special/Quality requirements:

Evaluation may take a while. However, submit time should take less than 5 seconds

5.1.1.4 Use Case: Create Group

Participating actors: Student

Pre-conditions:

- Student who creates the group must not be in a group
- Maximum number of group that can be created must not be reached

Post-condition:

- New group with the minimum number of members is created

Entry condition:

- Student is on the "Create Group Page"

- Group needs to contain at least two members to be created.

Exit condition:

- Student submits the group and leaves the page.

Flow of events:

- Student opens the “Create Group Page”
- Student manages requests of others to join
- Student sends invitations to others to join until group is full
- Student leaves the page

5.1.1.5 Use Case: Manage Requests

Participating actors: Student

Pre-conditions:

- There should be request to join to the group

Post-condition:

- The student is accepted or rejected

Entry condition:

- A student sends a request to join the group
- A member of the group manages the request

Exit condition:

- The request gets handled
- Student accepts or rejects the request and leaves the page

Flow of events:

- A student who wants to join a group sends a join request to the group
- A student from the group manages the request
- A student from the group accepts or rejects the request

5.1.1.6 Use Case: Invite Members

Participating actors: Student

Pre-conditions:

- A group must be created
- There must be room for a new member

Post-condition:

- An invite is sent to the student

Entry condition:

- A student of an existing group should be in the group page

Exit condition:

- Invite is sent and student leaves the page

Flow of events:

- A student from an existing group is on the group's page
- The student of the group selects and sends an invitation to another student
- If the other student accepts the invitation they are added as the member of the group

5.1.1.7 Use Case: Max number of groups reached

Participating actors: Student

Pre-conditions:

- A student must be trying to create a new group

Post-condition:

- Student can not create a new group

Entry condition:

- Student should try to create a new group

Exit condition:

- Student leaves the "Create Group Page" without creating a new group

Flow of events:

- A student tries to create a new group
- Their create request gets rejected because there is no room for a new group

5.1.1.8 Use Case: Discuss Evaluations

Participating actors: Student

Pre-conditions:

- An evaluation must be submitted

Post-condition:

- Two or more students discuss about the evaluation
- Group may resubmit their work

Entry condition:

- A student evaluate a group
- A group member starts a discussion at will

Exit condition:

- Students leaves the discussion

Flow of events:

- An evaluation is submitted
- Members of the group want to discuss about the evaluation
- Students may discuss about the evaluation on the Slack
- Group may resubmit their work based on the discussion

5.1.1.9 Use Case: Resubmit

Participating actors: Student

Pre-conditions:

- An evaluation must be submitted
- Group must discuss about the submission

Post-condition:

- Group resubmit their work

Entry condition:

- A member of the group must be in the submission page

Exit condition:

- Student resubmits the work and leaves the page

Flow of events:

- Group gets an evaluation about their work
- One or more members of the group discuss about the evaluation
- Based on the discussion a member of the group resubmits the work with changes

5.1.1.10 Use Case: Post a response

Participating actors: Student

Pre-conditions:

- An evaluation must be submitted

Post-condition:

- A member of the group posts a response about the evaluation

Entry condition:

- A member of the group responds to an evaluation

Exit condition:

- A member of the group submits their response to evaluation and leaves the page

Flow of events:

- A student submits an evaluation about group's work
- A member of the group shares a response for the evaluation
- No discussion needed

5.1.1.11 Use Case: Start Slack Discussion

Participating actors: Student

Pre-conditions:

- An evaluation must be submitted
- A group member must be start a discussion about the evaluation

Post-condition:

- No post conditions

Entry condition:

- A group member starts a discussion on Slack channel
- Students redirected to Slack channel

Exit condition:

- Students leave the Slack channel

Flow of events:

- A student submits an evaluation about a group's work
- A member of the group wants to discuss about the evaluation
- A member of the group starts a Slack discussion
- Two or more students discuss about the evaluation

5.1.1.12 Use Case: Create Course

Participating actors: Instructor

Pre-conditions:

- Course must not exist in the system

Post-condition:

- Course with the specific code is created

Entry condition:

- Instructor should be in the "Create Course" page

Exit condition:

- Instructor goes to another page

Flow of events:

- Instructor goes to "Create Course" page
- Instructor specifies the course name
- Instructor enters the course code
- Instructor describes the course

- Instructor specifies an entry code for students to enroll
- Student assigns one or more Graders for the course at will

Special/Quality requirements:

Instructor should be able to complete the events within a minute

5.1.1.13 Use Case: Create Project

Participating actors: Instructor

Pre-conditions:

- Specific project must not exist in the course page

Post-condition:

- Project is created and ready to enroll

Entry condition:

- Instructor should be in the “Create Project” page

Exit condition:

- Instructor goes to another page

Flow of events:

- Instructor selects the course which they want to create a project
- Instructor specifies the name of the project
- Instructor describes the project
- Instructor specifies the programming language of the project at will
- Instructor specifies the tools for the project
- Instructor specifies the group size
- Instructor may create random groups if they want to

Special/Quality requirements:

Instructor should be able to complete the events within a minute

5.1.1.14: Use Case: Assign Students to Groups

Participating actors: Instructor

Pre-conditions:

- There must be students who did not join a group

Post-condition:

- Students are assigned to random groups

Entry condition:

- Instructor should be in the “Groups” page

Exit condition:

- Instructor goes to another page

Flow of events:

- Instructor goes to “Groups” page
- Instructor may add groups manually
- Instructor may create groups randomly
- Instructor may assign students to groups manually

5.1.1.15 Use Case: Create Assignment

Participating actors: Instructor

Pre-conditions:

- The specific assignment must not exist

Post-condition:

- Assignment is created with specific deadline

Entry condition:

- Instructor should be in the “Create Assignment” page

Exit condition:

- Instructor goes to another page

Flow of events:

- Instructor specifies the title of the assignment
- Instructor describes the assignment
- Instructor gives a specific deadline for the submission
- Instructor selects the progress of the assignment
- Instructor shares the assignment with students

Special/Quality requirements:

Instructor should be able to complete the events within a minute

5.1.1.16 Use Case: Submit Assignment

Participating actors: Student

Pre-conditions:

- Deadline of for the submission must not be reached

Post-condition:

- Assignment is submitted for grading

Entry condition:

- Student should be in “Assignment” page for students

Exit condition:

- Student goes to another page

Flow of events:

- Student goes to specific assignment page
- Student drags the submission file to proper area before deadline
- Student submits the assignment

5.1.1.17 Use Case: Grade Assignment

Participating actors: Instructor, Grader

Pre-conditions:

- Specific assignment must be submitted by the student
- The assignment must not be graded before

Post-condition:

- Student's overall grade is effected

Entry condition:

- User must be grading page for assignment

Exit condition:

- User goes to another page

Flow of events:

- User grades each criteria out of 10
- User comments gives feedback as comment at will
- User submits the grade of the assignment

Special/Quality requirements:

Grading process may take a while. However, users should be able to submit grades according to criteria within 5 minutes

5.1.1.18 Use Case: Send Message to Another User

Participating actors: Student, Instructor, Grader

Pre-conditions:

- No Pre conditions

Post-condition:

- The message is sent to specified user
- Receiver gets a notification

- Receiver can see the message in their inbox

Entry condition:

- User must be in the “Message” page

Exit condition:

- User goes to another page

Flow of events:

- User specifies the receiver's name
- User specifies the title of the message
- User enters their message
- User sends the message to the receiver

5.1.1.19 Use Case: Join Project

Participating actors: Student

Pre-conditions:

- Student must not join a project in the same course

Post-condition:

- Student works in a group who works on that project

Entry condition:

- Student sends a request to work on a specific project

Exit condition:

- Student goes to another page

Flow of events:

- Student chooses an assigned project
- Student sends a request to work on the project
- Instructor assigns the student to an available group who works on that project

5.2 Object and Class Models

5.2.1 Package Diagram

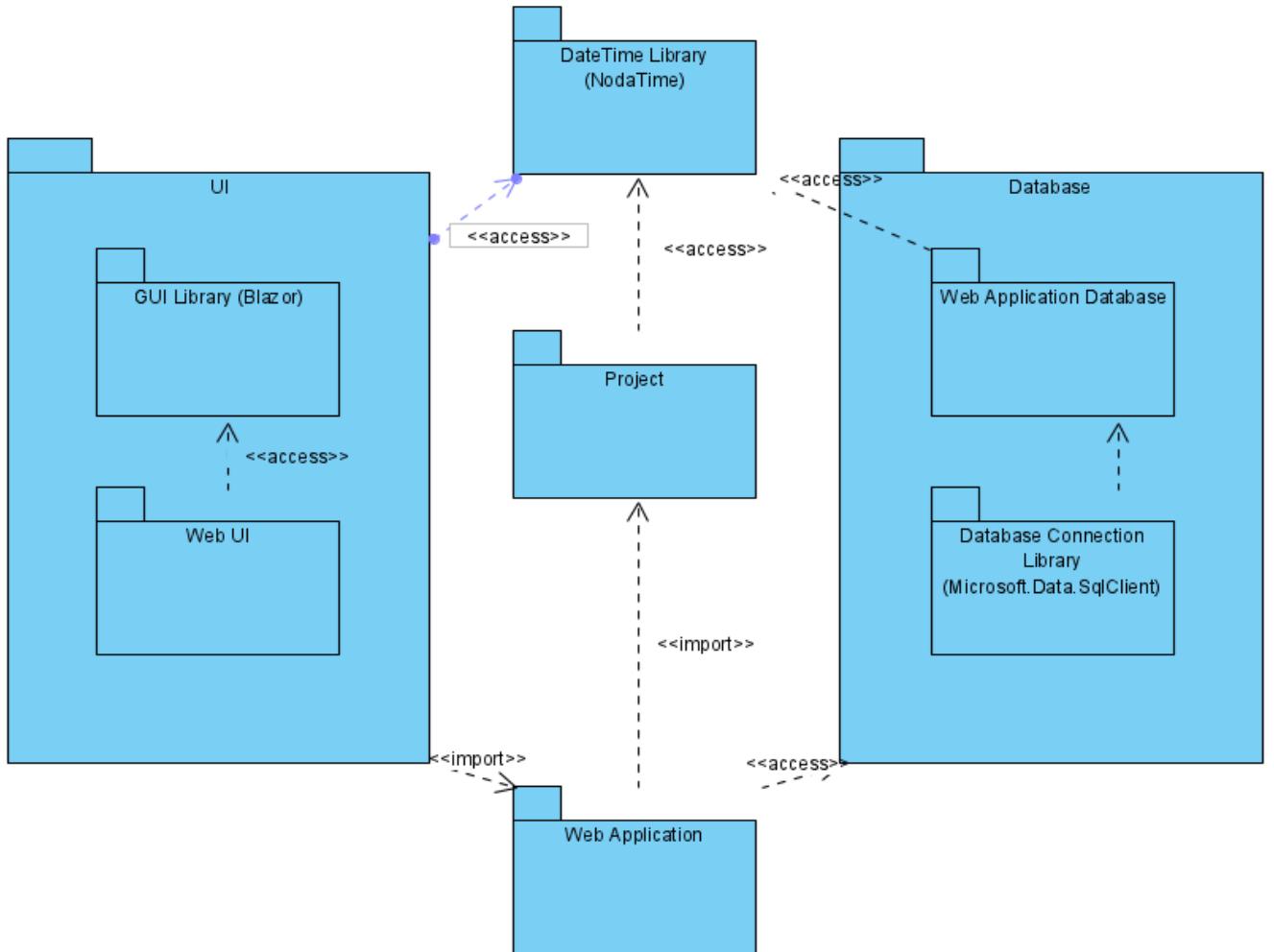


Figure 5.2.1 Package Diagram

For user interface it is thought that a library for gui might be beneficial especially to make eye appealing service. Thus, we might access some of Blazor Library functions to handle our ideas better in our web ui package. For data handling we thought that a database might ease the process. Hence, we might call from Microsoft.Data.SqlClient library. For time management we have our date service. In case of a better solution NodaTime library might help us. Project has the forthcoming capabilities, and the web application is where our ideas become alive.

5.2.2.1 User Student Grader Diagram

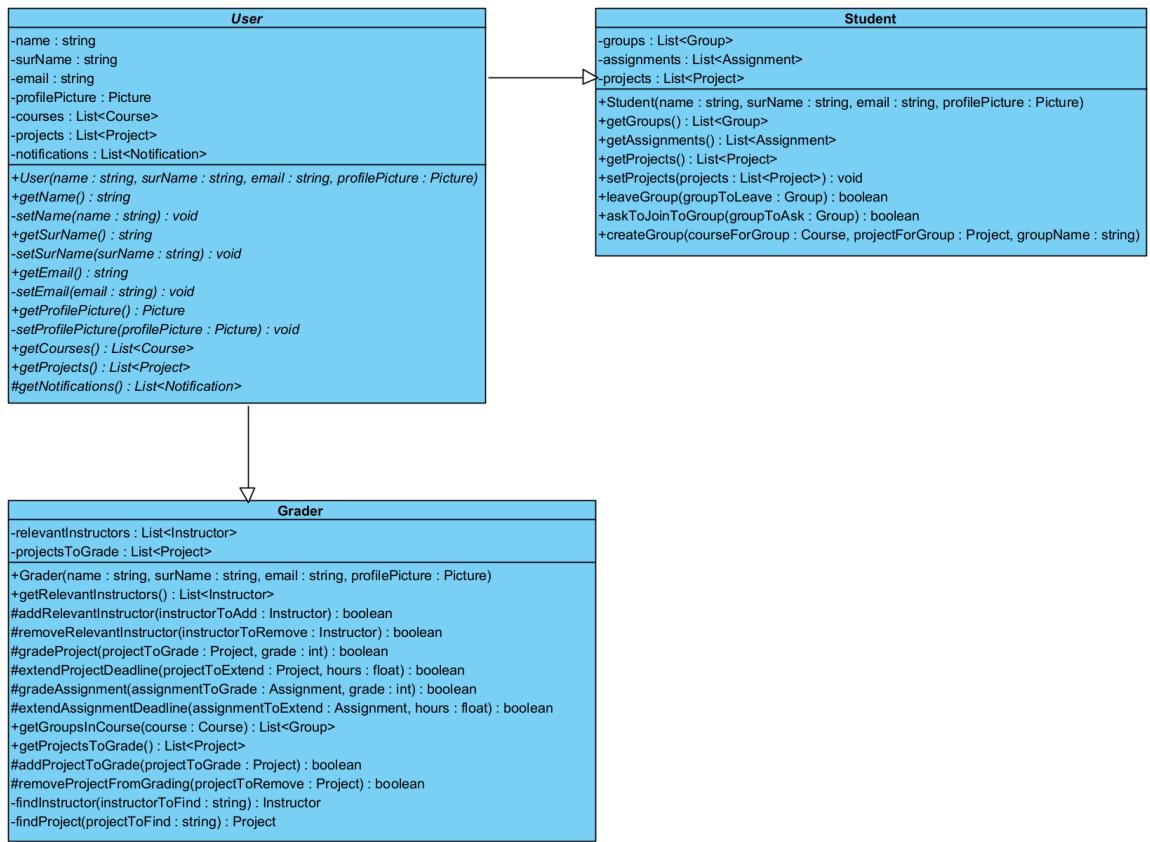


Figure 5.2.2.1 User Student Grader Diagram

In the system we have an abstract class called User which is basically any user. That has a name, surname, email address, set of courses, set of projects, profile picture, notifications, and a set of messages. From the user, we have the student class which has a set of groups to keep track of students distinct groups from other courses, a set of assignments to help students get their grades easily, and a set of projects to help them to keep track of what is gStudent submits their group if room for a group is available going on in each project.

Grader is the instructor assigned masters or phd students having some capacity to grade students while not having the unrestricted instructor capabilities.

5.2.2.2 User Instructor Relationship



Figure 5.2.2.2 User Instructor Relationship

Instructor has a set of graders that are assigned by them, and the given course list. Nearly everything that is related to them is under their control. Assigning homework, extending deadlines, randomly assigning students to groups are examples of what instructors can do within seconds.

5.2.2.3 User Course Relationship

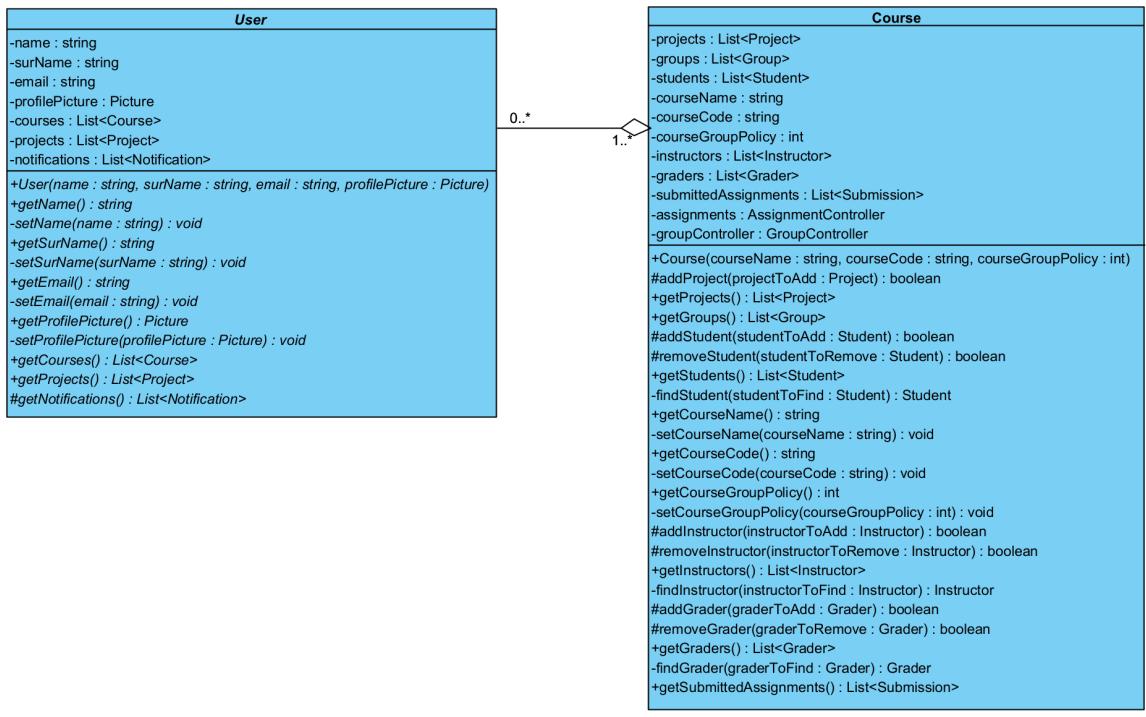


Figure 5.2.2.3 User Course Relationship

User and course relationship is an aggregation, if all users removed from the course then there is at least zero users in, and there might exist a user with zero courses. Course has a course name, course code, a set of instructors, graders and students as well as assignments, submitted documents, and groups formed for projects.

5.2.2.4 Course Project Relationship

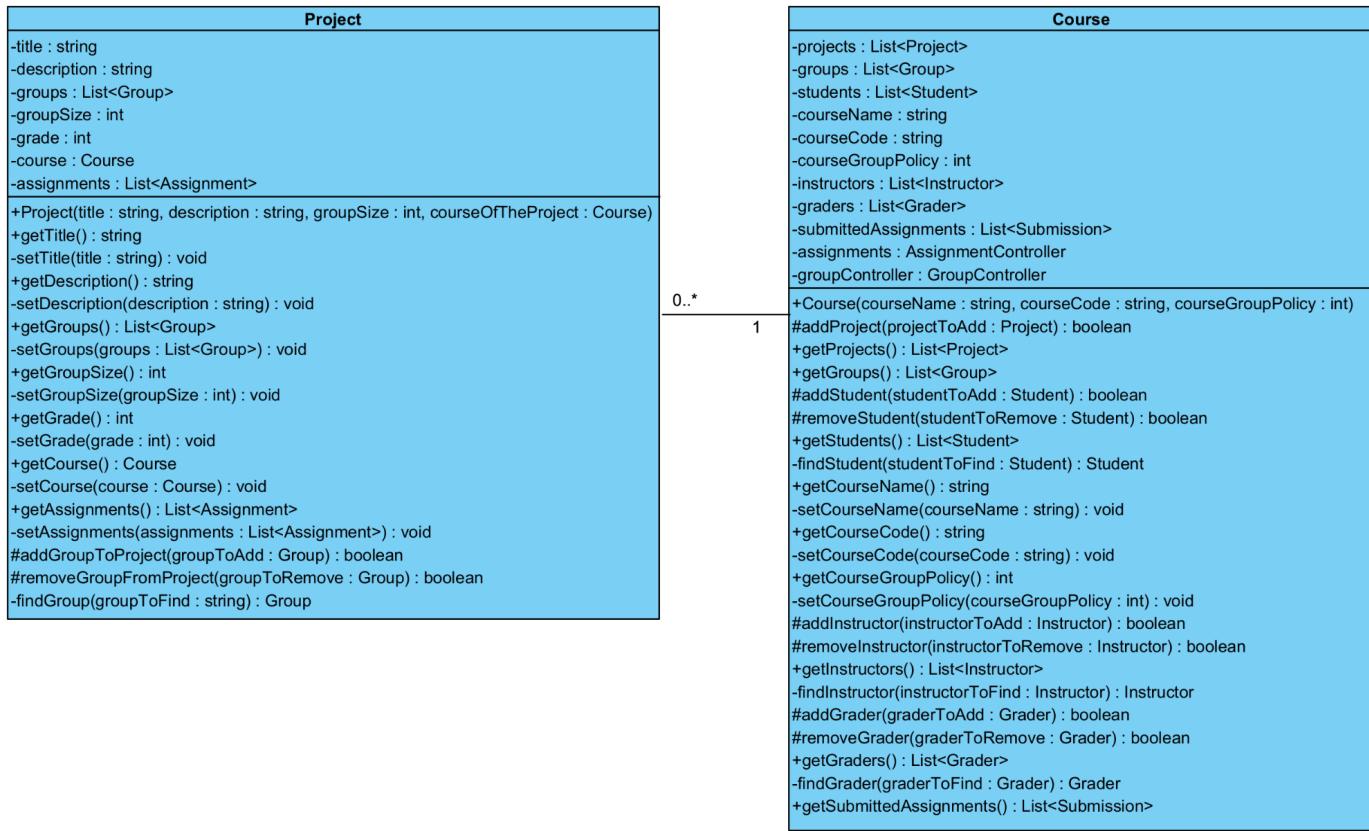


Figure 5.2.2.4 Course Project Relationship

Course has a name, course code, and a set of instructors, graders, students, assignments, submitted documents, and groups. Course group policy is to limit the group related issues. Some functions are private to make the course more secure and the relationship between a course and a project is thought to be a multiplication where a project requires a course while the course has zero or more projects.

5.2.2.5 Group Project Relationship

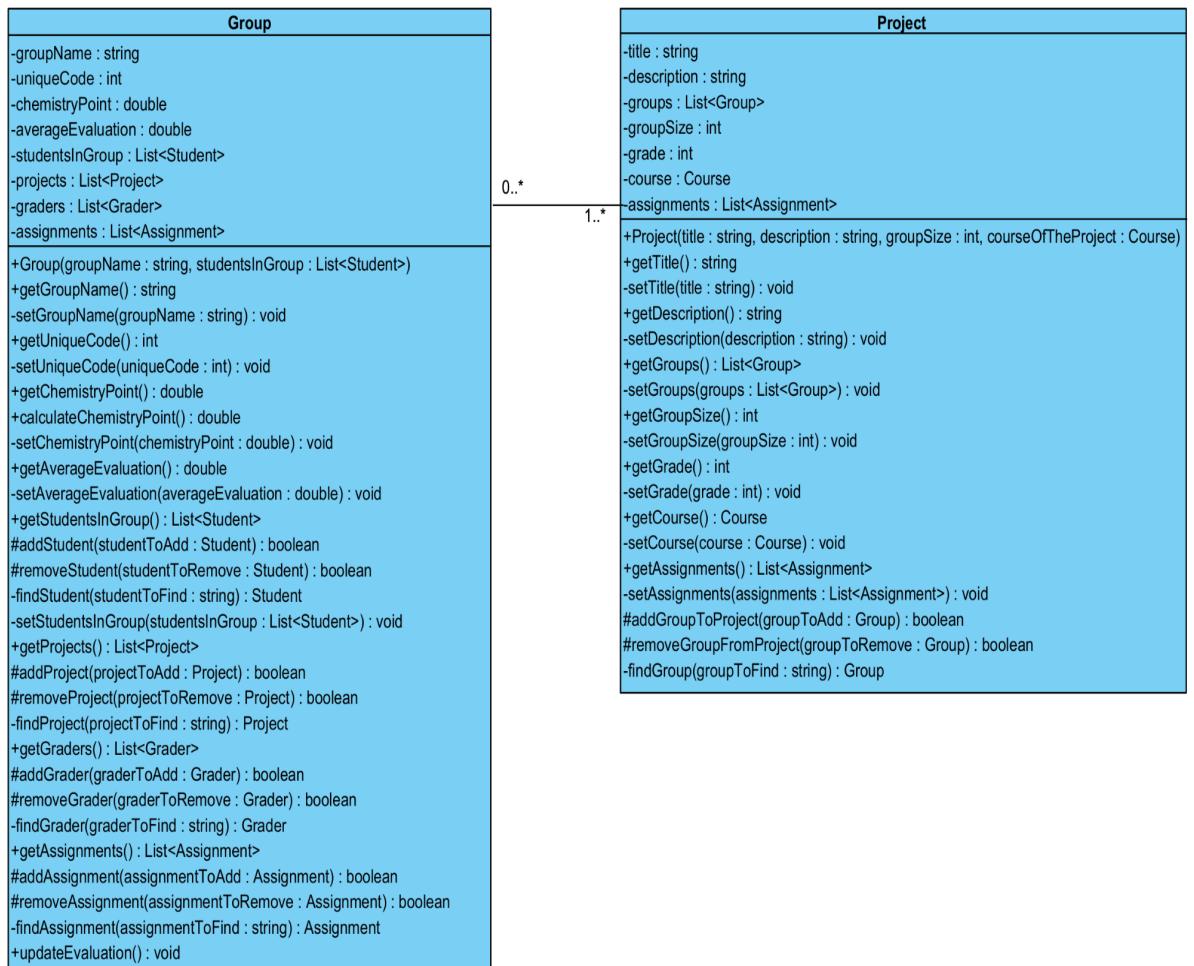


Figure 5.2.2.5 Group Project Relationship

Group is a very distinct set of students that are in the same course. Set of students, and set of projects that the same group of students done can be accessible within the group. Group name and code is what makes them distinguishable from the groups. Grader is who is supposed to grade, or mentor that group, and the assignments are the set of assignments that are done by that group. Distinction between set of projects and set of assignments is, for example CS 315 requires two projects that should be done by the same group while CS 461 requires six assignments and one project that should be done by the same group. CS 315 projects are kept in the projects and the CS 461 assignments are kept in the assignments. Chemistry score is a number in an interval which may be used to get the information about how well the members of the other group fit in each other. Students might have the same instinct to help others thus, they might have higher chemistry scores, as if they act similarly. Evaluations of the groups are kept. It is thought that an instructor may want to get evaluations after every group activity, or just at the end of the semester, that is what we aimed to provide. Evaluations are just as close as a function call.

5.2.2.6 Comment Review Response Submission Relationship

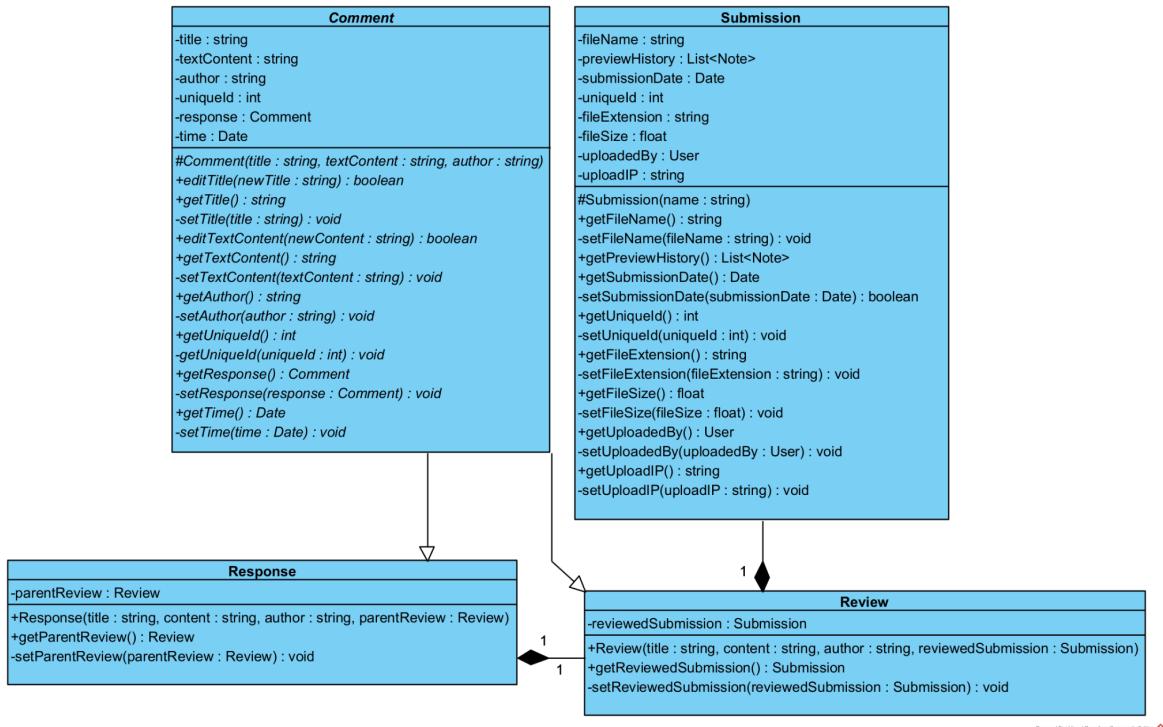


Figure 5.2.2.6 Comment Review Response Submission Relationship

Comment is a blueprint for any sort of Review or Response. Review is a collection of submissions. Response is a collection on Reviews.

5.2.2.7 Evaluation Review CompletedSubmission Submission Note Relationship

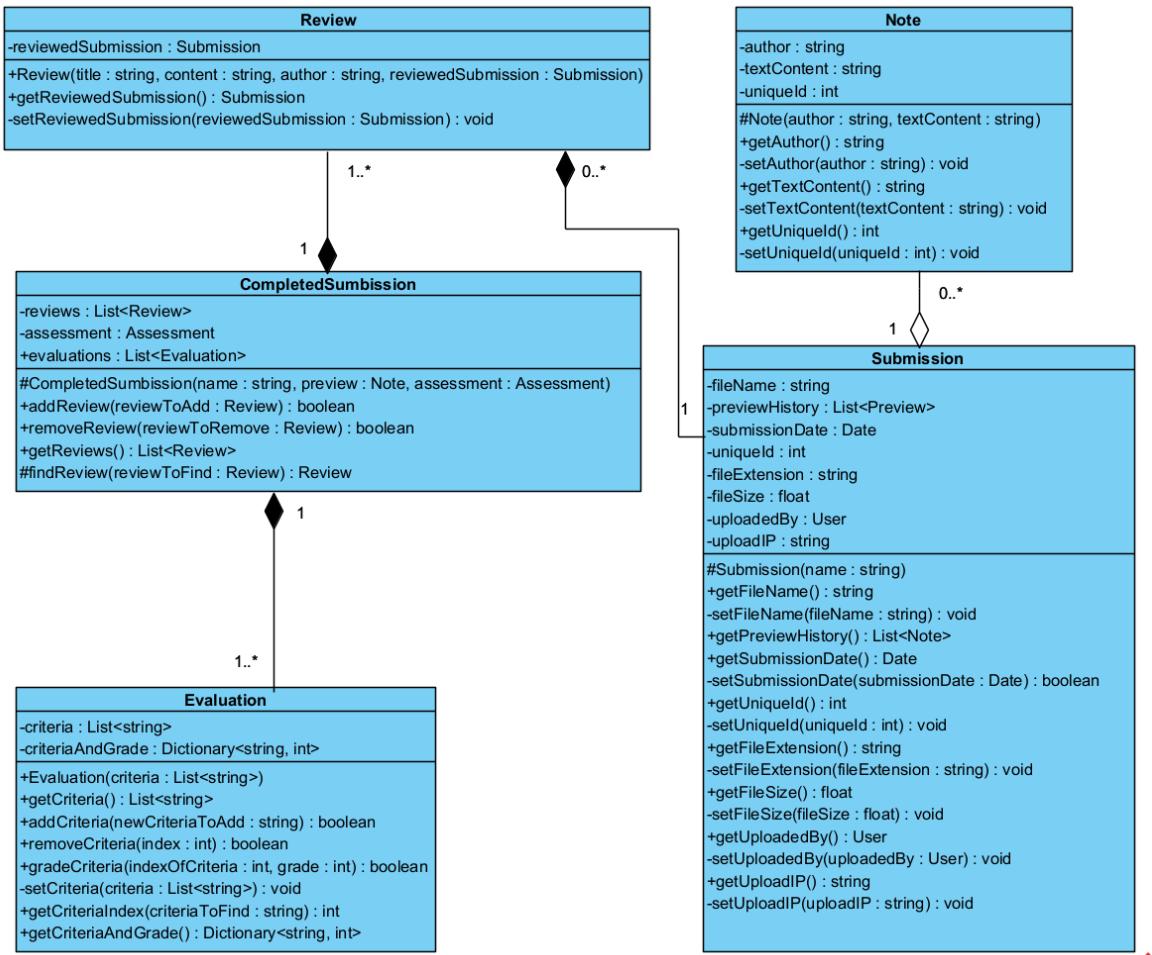
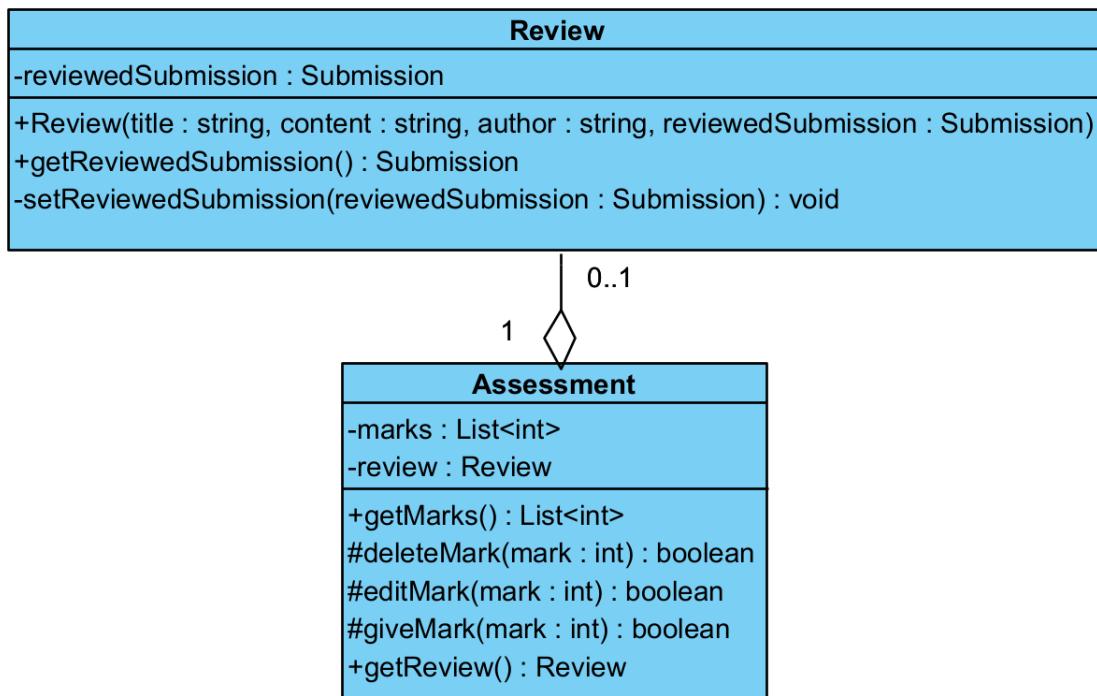


Figure 5.2.2.7 Evaluation Review CompletedSubmission Submission Preview Relationship

Reviews and Evaluations are part of CompletedSubmissions. Each Submission has a collection of Notes, from which Responses and Reviews inherit, the top Note points at its children Reviews and Responses. Submission may include a collection of Notes, a class that Reviews and Responses inherit because of their similar attributes.

5.2.2.8 Review Assessment Relationship



Powered By: Visual Paradigm Community Edition

Figure 5.2.2.8 Review Assessment Relationship

Assessment is a collection of Reviews however; an Assessment does not require a Review.

5.2.2.9 Group Course Diagram

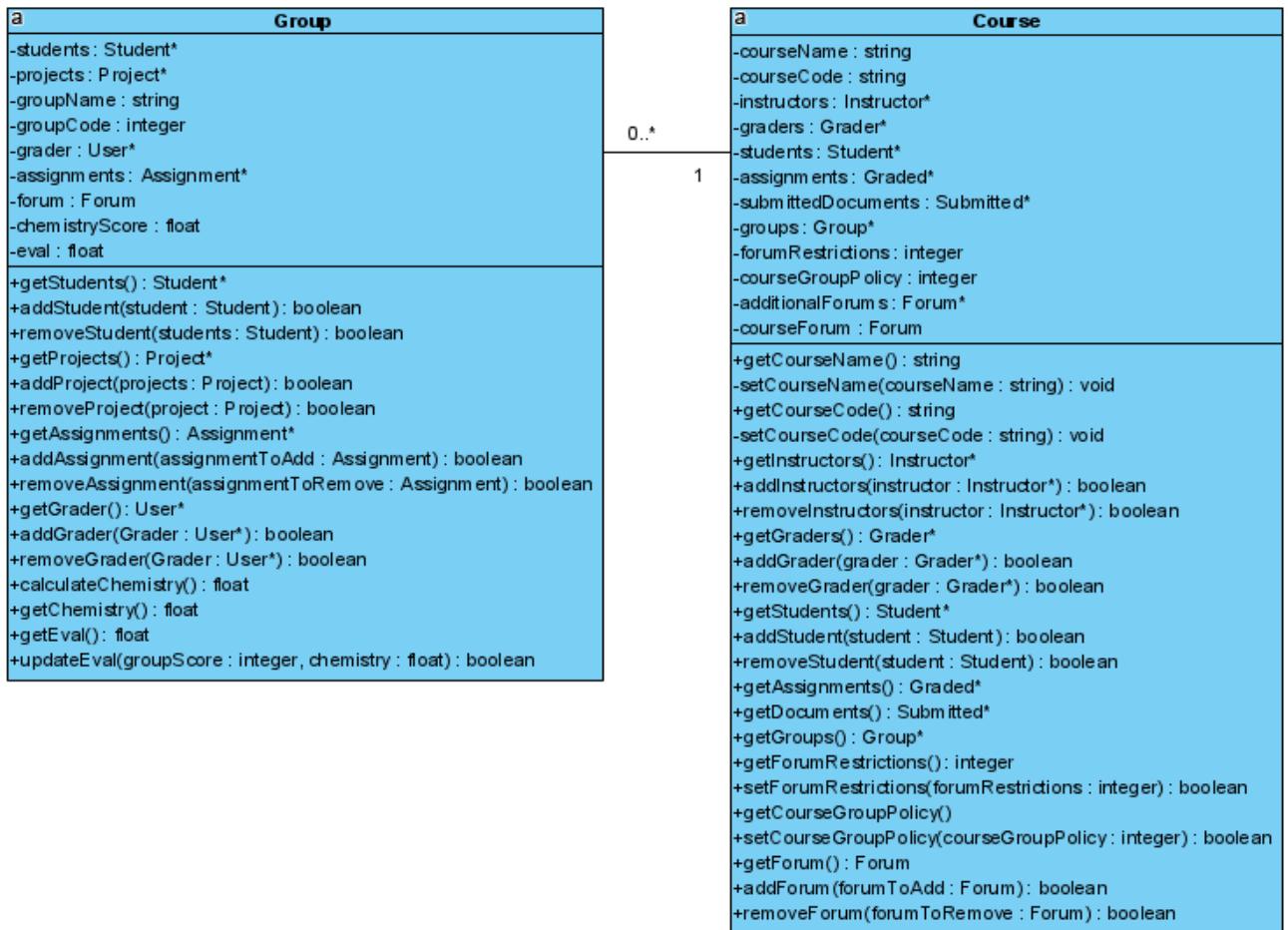


Figure 5.2.2.9 Group Course Diagram

Courses and groups in that course have their multiplication as a course may have zero or more groups while a group should have only one course.

5.2.2.10 Student Group Diagram

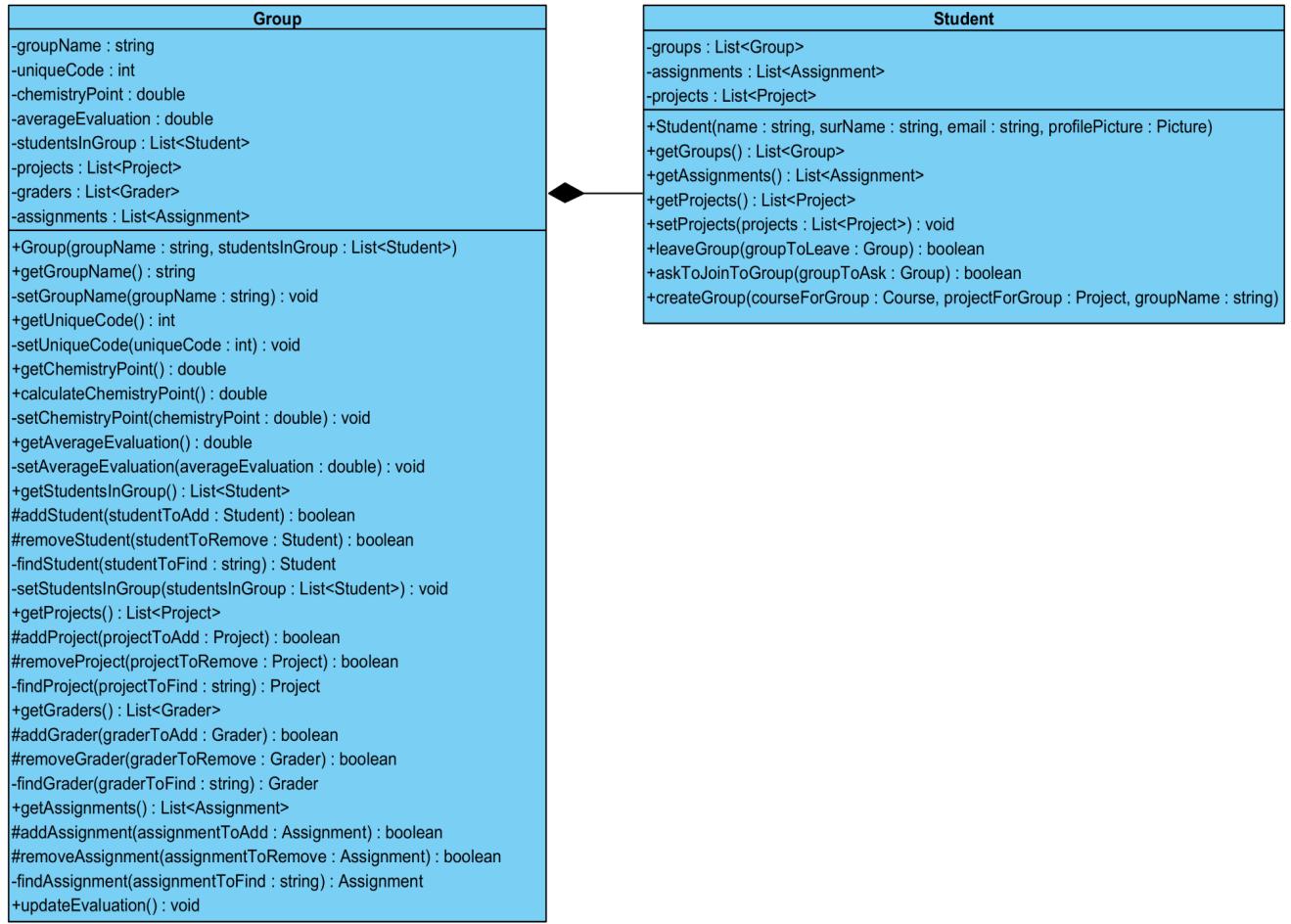


Figure 5.2.2.10 Student Group Diagram

Students can form a group as long as they obey the limitations of the instructor.

5.2.2.11 Manager Class

Manager
#userController : UserController #courseController : CourseController -users : List<User> -graders : List<Grader> -instructors : List<Instructor> -students : List<Student> -courses : List<Course> +getUserController() : UserController -setUserController(userController : UserController) : void +getCourseController() : CourseController -setCourseController(courseController : CourseController) : void +getGraders() : List<Grader> -setGraders(graders : List<Grader>) : void +getInstructors() : List<Instructor> -setInstructors(instructors : List<Instructor>) : void +getStudents() : List<Student> -setStudents(students : List<Student>) : void +getCourses() : List<Course> -setCourses(courses : List<Course>) : void

Powered By Visual Paradigm Community Edition 

Figure 5.2.2.11 Manager Class

The manager contains all the courses and users. It has methods to add or remove courses, student and grader and also a method to get all courses for a specific user and check for an existing user.

5.2.2.12 Notification Class

Notification	
-title : string	
-text : string	
-date : Date	
-actionSource : User	
-actionTarget : List<User>	
+getTitle() : string	
-setTitle(title : string) : void	
+getText() : string	
-setText(text : string) : void	
+getDate() : Date	
-setDate(date : Date) : void	
+getActionSource() : User	
-setActionSource(actionSource : User) : void	
+getActionTarget() : List<User>	
-setActionTarget(actionTarget : List<User>) : void	

Powered By: Visual Paradigm Community Edition

Figure 5.2.2.12 Notification Class

The notifications have a sender, date and content and get and set methods for all attributes

5.2.2.13 Date Class

Date
-day : int
-month : int
-year : int
-hour : int
-minute : int
-dateAsString : string
+getDay() : int
+setDay(day : int) : void
+getMonth() : int
+setMonth(month : int) : void
+getYear() : int
+setYear(year : int) : void
+getHour() : int
+setHour(hour : int) : void
+getMinute() : int
+setMinute(minute : int) : void
+getDateAsString() : string
- setDateAsString() : void

Figure 5.2.2.13 Date Class

The date is in the normally used format with get and set methods for every attribute.

5.3 Sequence Diagram

5.3.1 Course Creation

In this diagram we show the sequence of the interactions between the instructor and the system to create a new course. As explained before the instructor has the authority to create a course, first we check if the course does not already exist, before creating it. After creating the course the instructor can perform several operations like adding students, creating projects, adding graders to the course and assigning students randomly to groups. When Instructor decides to create a project the call is sent to the course where the project is created and added to it.

While creating a course several operations can be performed, the system check for any possible issues like adding a grader that does not exist or removing a student that is no in the course. Almost all of the operations are performed directly on the new course instance, however, while creating a course the instructor can create a project and assign students to random groups, in these cases the interactions will be also with the project instance newly created.

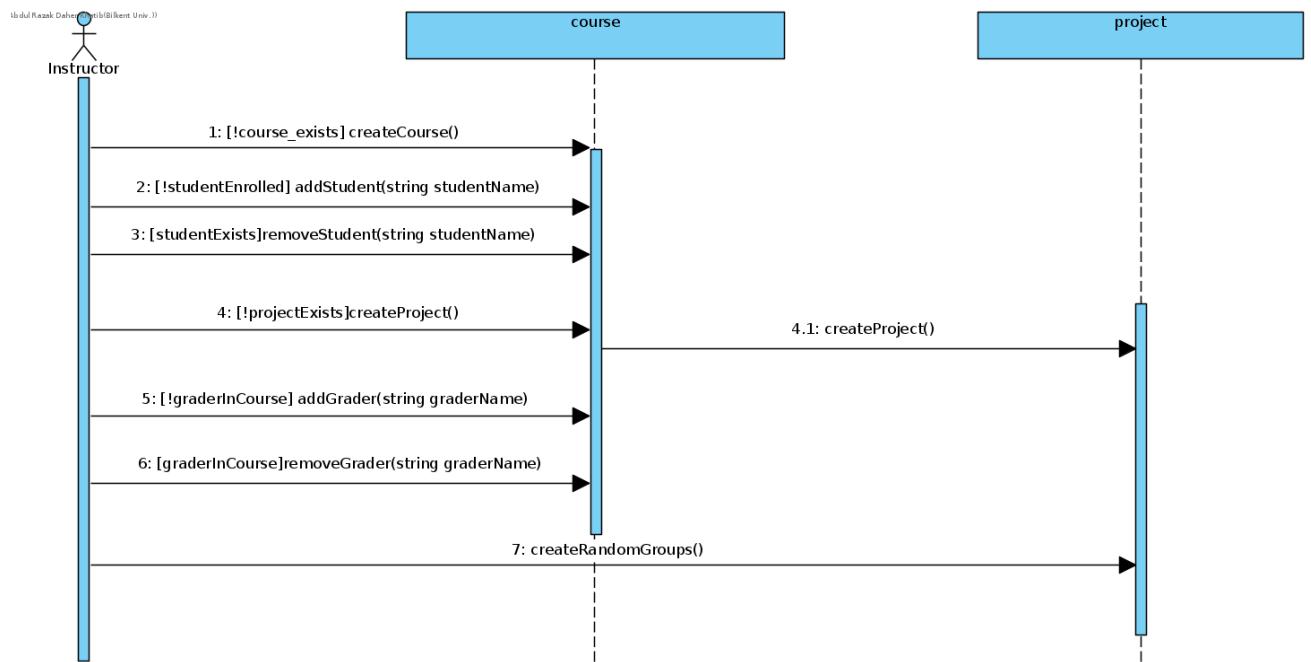


Figure 5.3.1: Instructor creating a course sequence diagram

5.3.2 Grader Grading Submission

This diagram shows the brief interactions of a grader user with the system. Graders can grade an assignment, which includes adding a review and leaving a grade. The grader can also respond to a response left by a student to the review. Also the grader can edit her assessment of the submission or endorse a review made by other students, which might save the time for the grader and possibly make students earn marks through review each other.

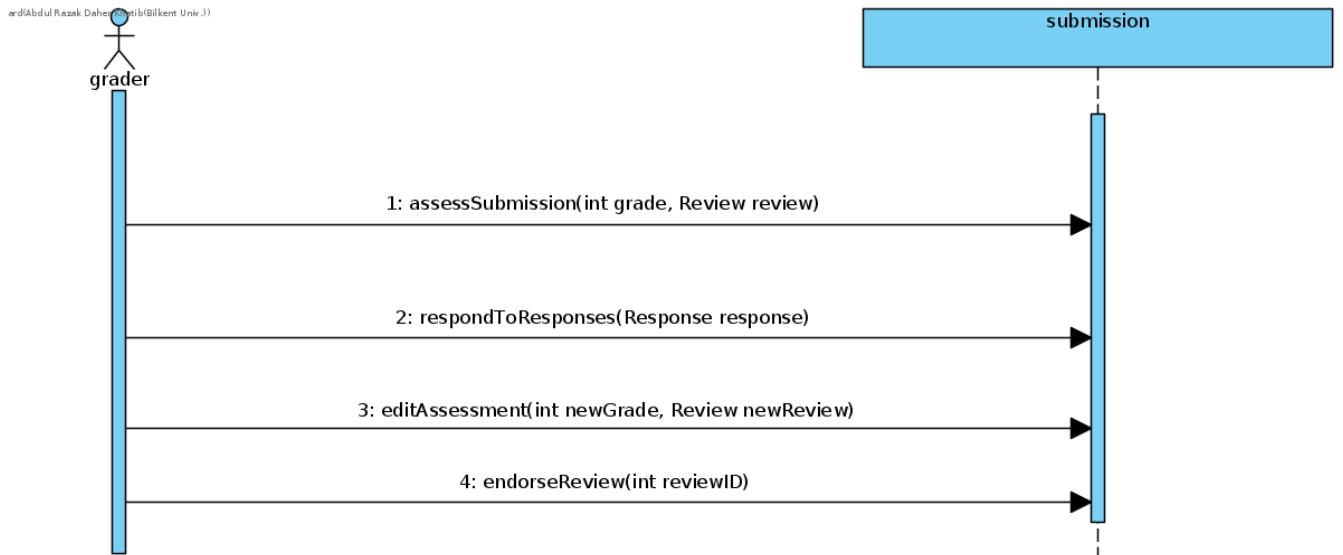


Figure 5.3.2: Grader grading submission sequence diagram

5.3.3 Group Formation

In this diagram we show the interactions of students while forming groups. In this diagram we can see the various operations performed while forming groups. At the beginning when a new group is requested to be added the project instance checks if the number of the group did not reach its maximum, for example if we have 30 students and groups size of 6, no more than 5 groups will be allowed, then if it did not reach the limit it creates a group and add it to the project and adds the student to the group newly created, if the number reached the maximum or if the instructor chooses the random assignment option then the student will not be able to create a group and is notified by an error message. Additionally, if the student chooses to send a request to join a group then if the option is enabled by the group then the user awaits the decision of the group, if accepted he is added to the group if not the user is notified of his rejection. One more option is receiving an invitation, if the group sends an invitation to a student in the project then she accepts it, she will join the group. We check if the user is not in a group first then await for her response and notify the group of that and add the student to the group if she accepts. The student also has the option to leave the group during the group formation period, as well as deleting the group in case she had the authority which will be given in case the student was the leader or the only member. In case the group was deleted it will also be removed from the project.

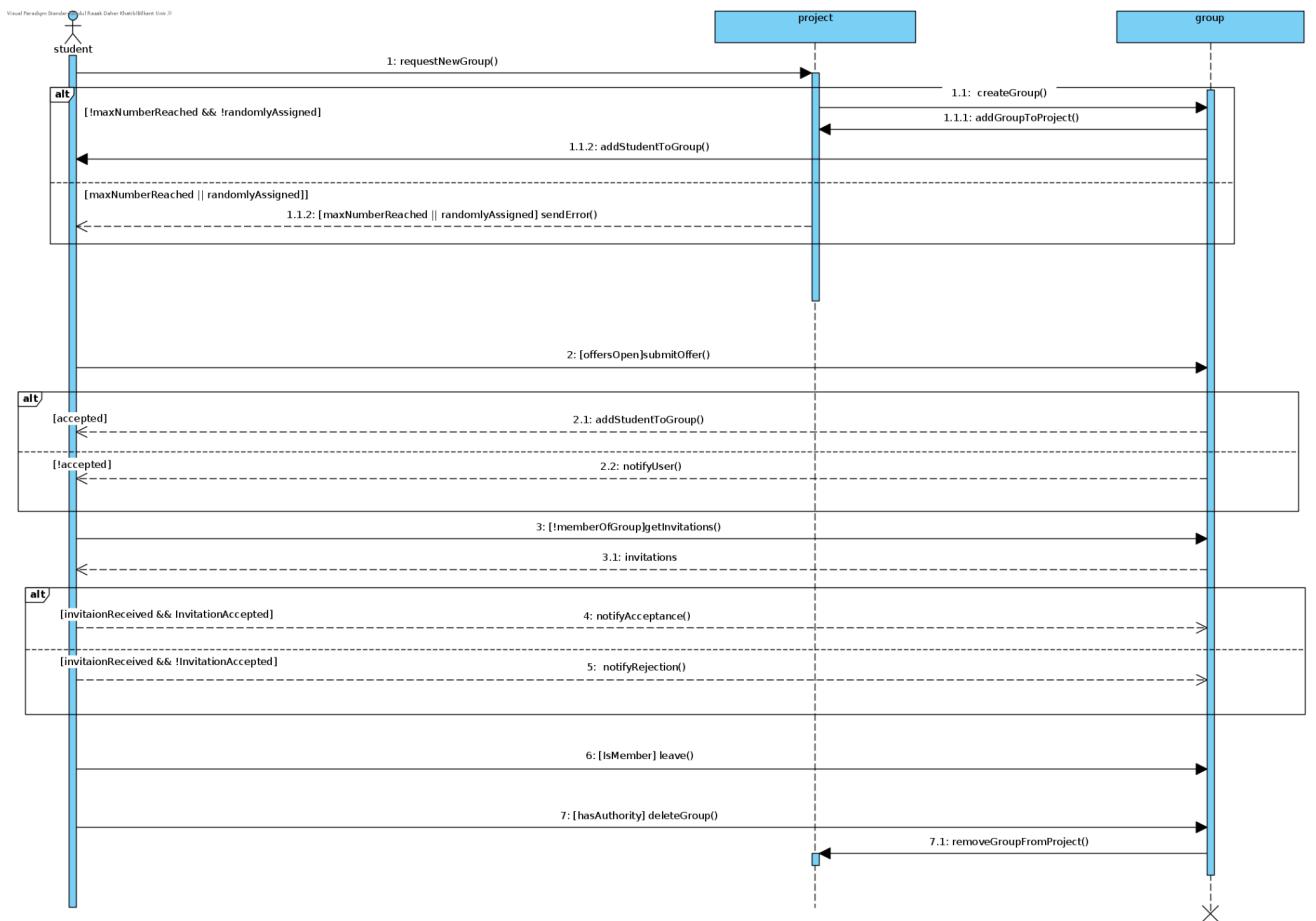


Figure 5.3.3: Group formation sequence diagram

5.4 Activity Diagram

5.4.1 Reviews from other groups' members

As seen in the figure below after a review is received from another groups' members, the student can either address it or not. If they decide not to address, the review activity is finished. However, if they choose to address it, they can either post a response on Peerview, which will simply add their comments under the responses to the review and notify the original poster, or start a discussion on slack which will take them to the specific slack channel where they can discuss the submission in detail. Moreover, after reviewing, the user can edit a review and submit it again. If they receive a follow-up to the , the same process for reviews will repeat; if not, the review activity would finish. Finally, there will also be a time check for reviews and if it has expired, the activity would finish.

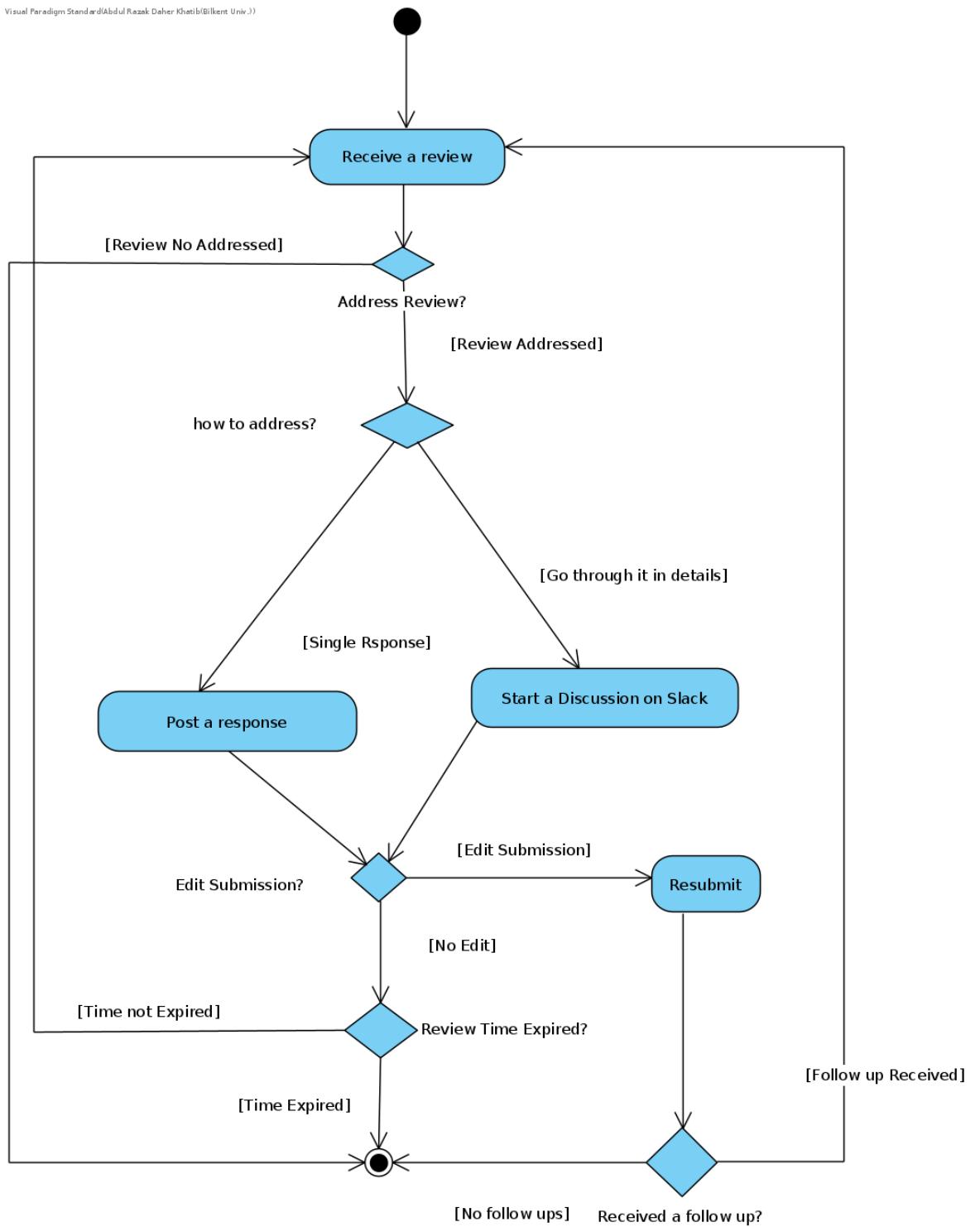


Figure 5.4.1: Reviews from other groups' members activity diagrams

5.4.2 Group Creation

For the group formation process that starts after projects are announced and open there are multiple ways of forming and joining groups. The instructor can choose to create groups and assign students randomly. If the instructor does not choose such a way the student would have three ways to join a group. The first one is to join by invitation, i.e., if the user is already invited to join a group, they can accept the request and become part of the group. The second way is to join by request; here, the user would be shown all the vacant groups and can apply to those. If they get accepted, they will become part of the group, if they get rejected, they can start over with any of the three ways to join a group. The third way is to create their own group where they can add members via sending invites and accepting requests. Once a group is full, it will be closed for invites and requests and thus by these three ways the group formation process terminates once all students are part of a group, this process can be shown in the figure below.

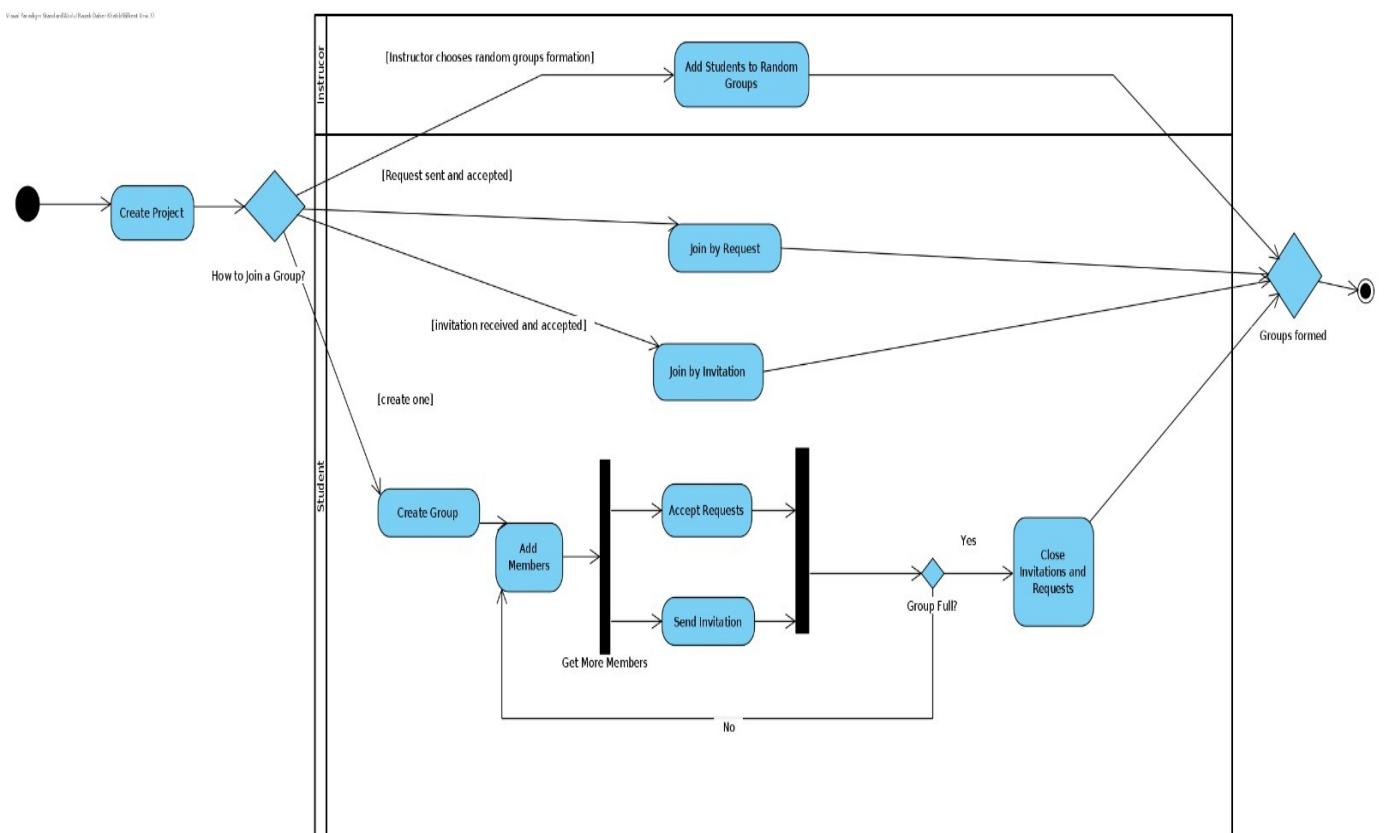


Figure 5.4.2: Group Creation

5.5 State Diagram

5.5.1 Submissions

Once an assignment is announced by the instructor or a grader and is open for submission the students have to submit the assignment. Afterwards two processes start at the same time; the grading and the evaluation by other groups' members. The grading can be done by multiple graders, after which the average is calculated and added. Once both of these states are finished the students can choose to edit their submission according to the feedback they received. Afterwards, they can go through the same process. Now for evaluation it is open again for students to review but for grading it is expected that the graders that did not grade yet can grade or for the same graders to grade again if they wish to, so if this feature is enabled it represents a form of a grace period where students have a period of time where they can edit their submission without the grade being final, if not enabled then they cannot edit the submission. In some cases a submission would have multiple cycles, we will connect these cycles to each other and show changes and comparisons in evaluations and grades to the students. There can be as many cycles of submissions as desired by the instructor, all of these cycles would be compared to each other and analyzed. All of this can be seen in the figure below.

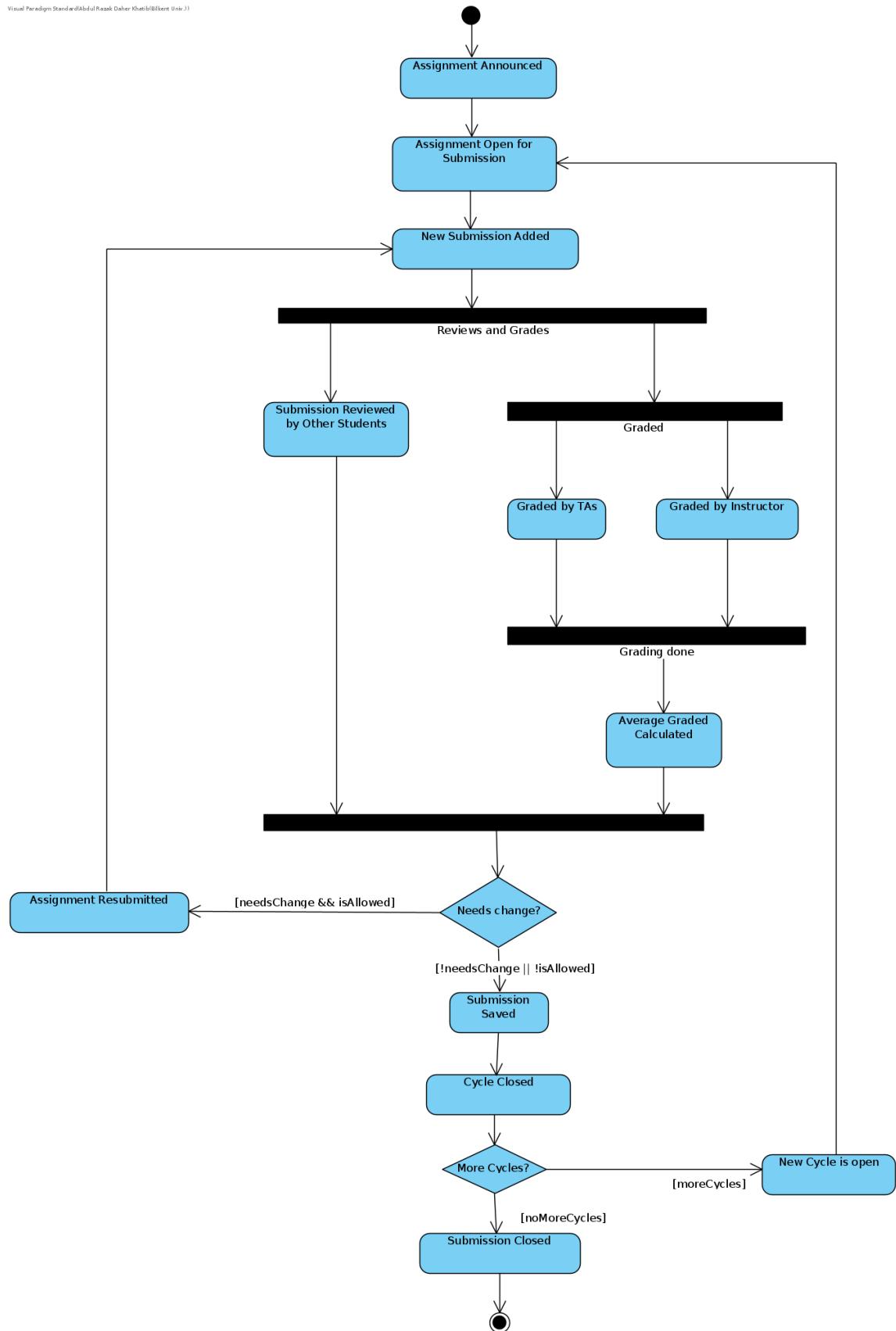


Figure 5.5.1: Assignments

5.5.2 Groupmates Evaluations

In this figure, we see how students evaluate their group mates. Once the project is starts and submissions are open and submitted, the assignment is evaluated and graded as shown in the activity diagram above, afterwards, once a cycle is closed, the students get to evaluate their group mates, the evaluations will be based on specific criteria and the students will have the option to leave a note with the grades. Now if the instructor allows it an “early warning” is sent to members with low evaluations that their group mates are not very happy with their contribution. Afterwards, the evaluations are analyzed and visualized. Finally, if there is another cycle the same process repeats except that this time the improvements are evaluated to show if the members worked hard enough to improve their performance, then afterwards the analysis and visualization is saved and added to the submission once there are no more cycles or iterations.

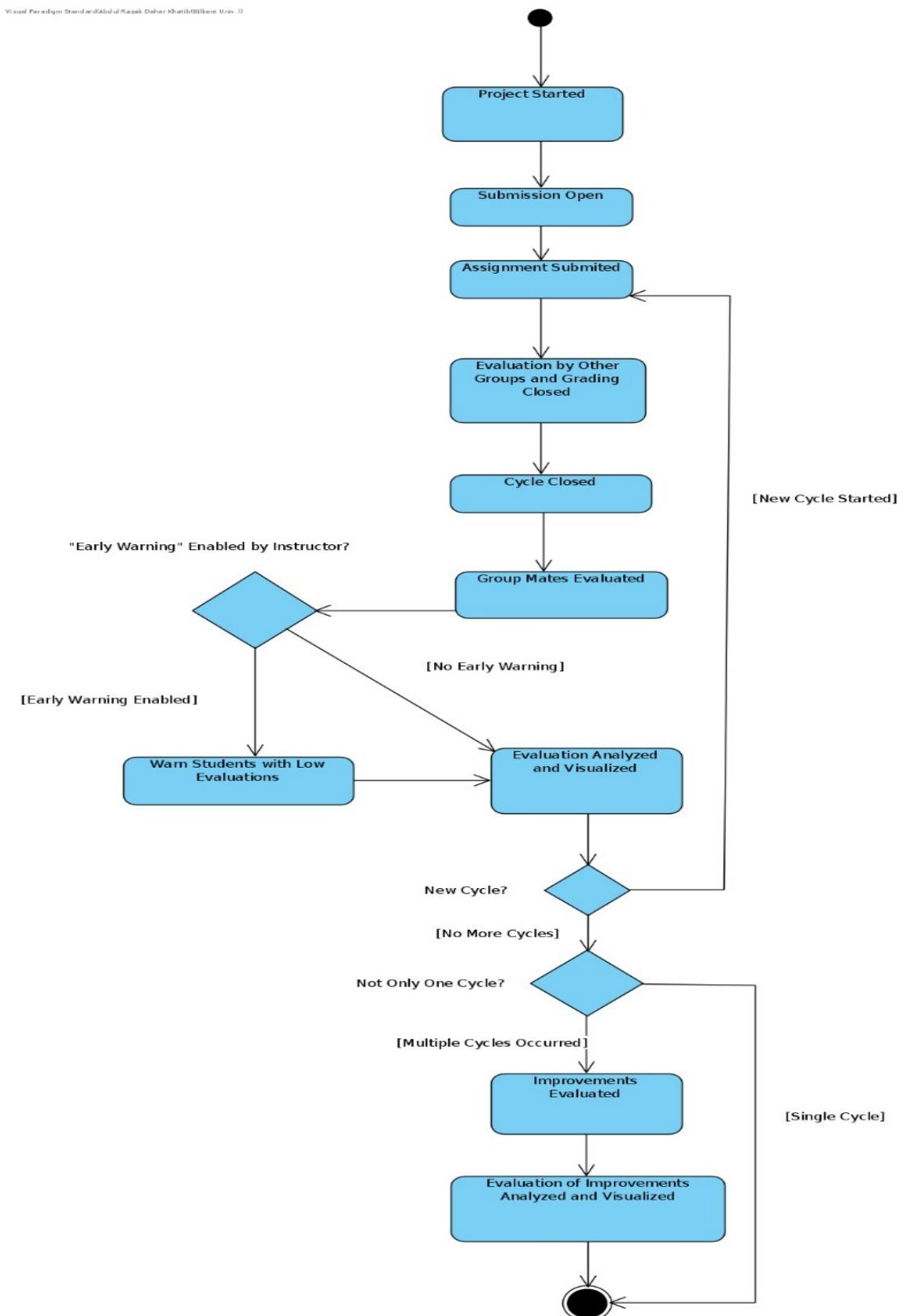


Figure 5.5.2: Groupmates evaluations

5.6 Screen Mockups

5.6.1 Login As Page

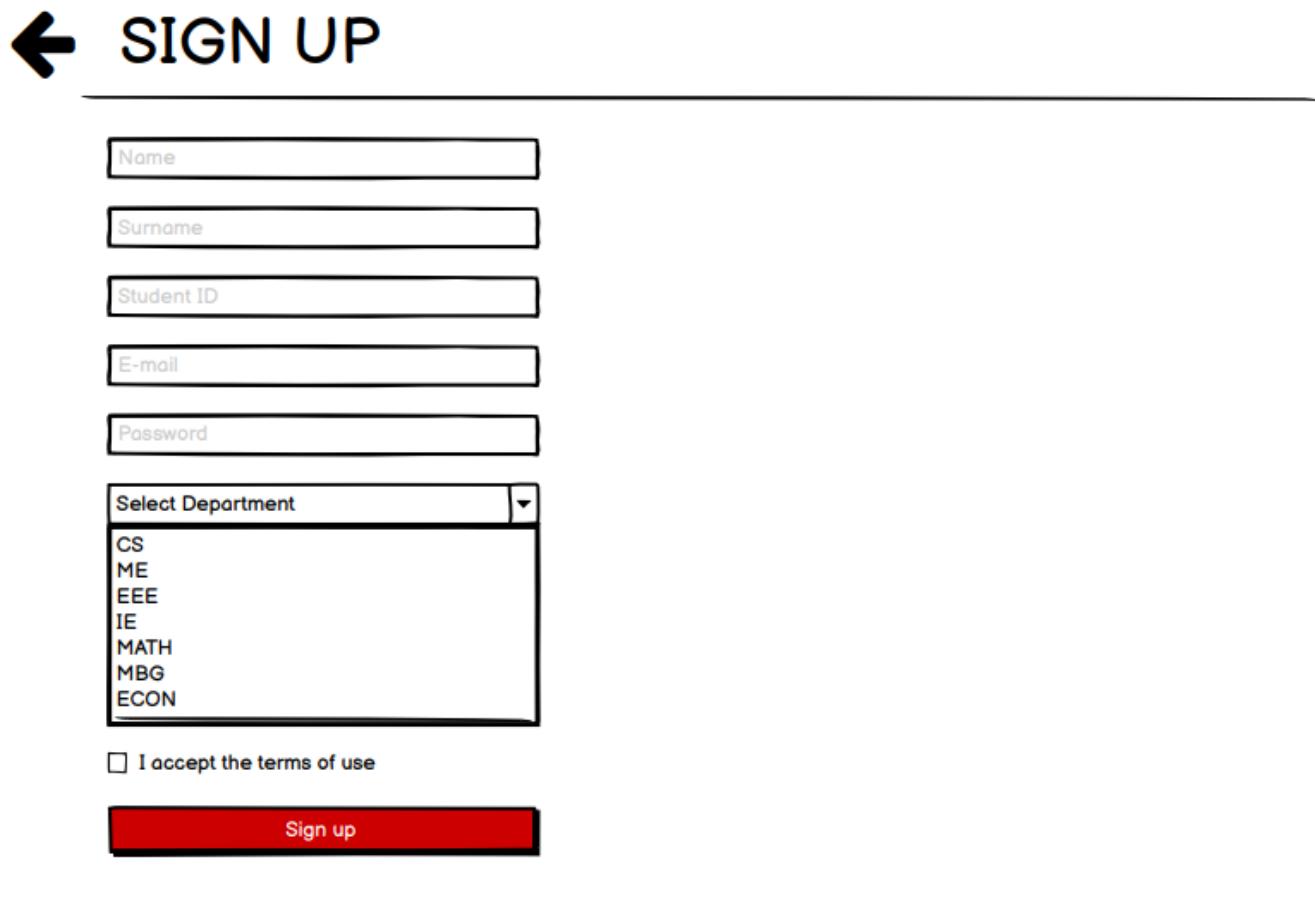


Figure 5.6.1: Login As Page

In Figure 5.6.1, the user is asked whether she is an instructor, a student or a teaching assistant. After that, the user is directed to the "Sign Up" page accordingly.

5.6.2 Sign Up Page

5.6.2.1 Sign Up Page For Students

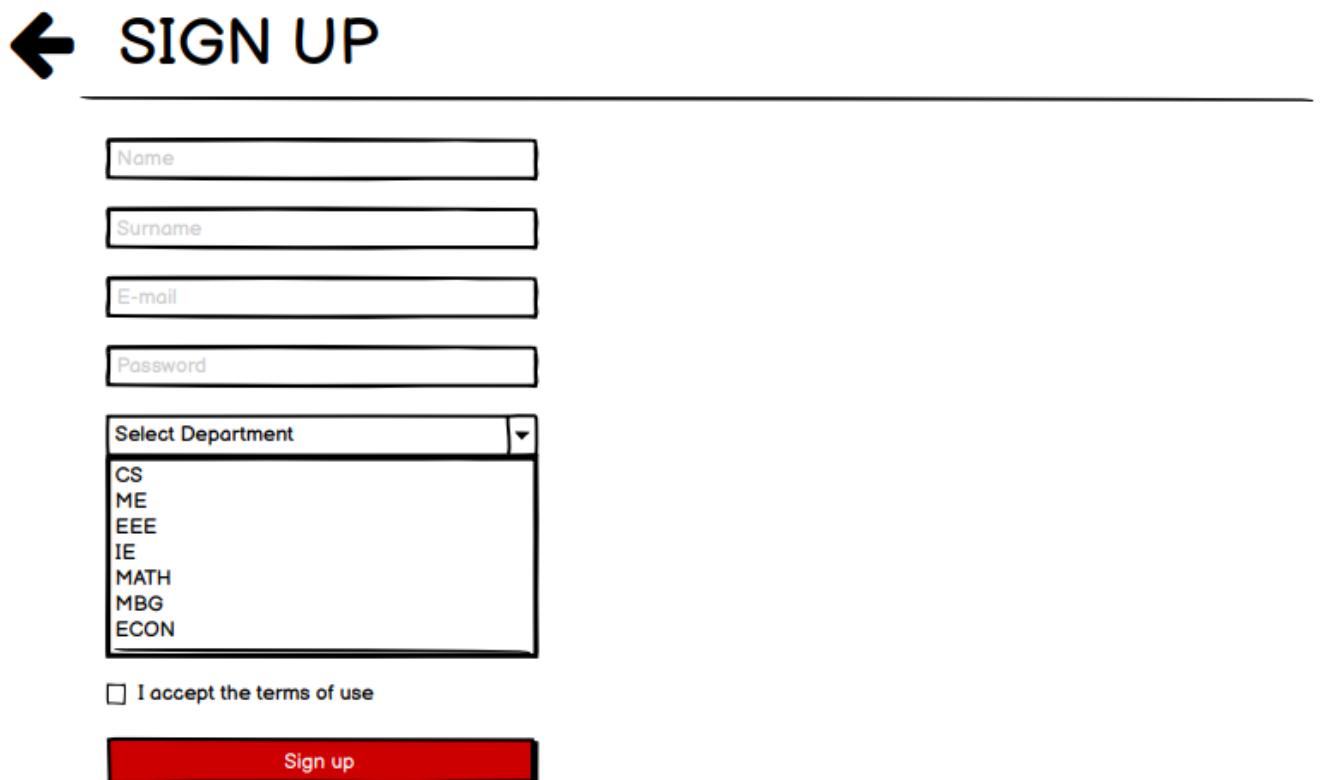


The image shows a sign-up form for students. At the top left is a black back arrow icon. To its right, the words "SIGN UP" are written in a large, bold, black sans-serif font. Below this is a horizontal line. The form consists of several input fields and a dropdown menu. On the left side of each input field is a label: "Name", "Surname", "Student ID", "E-mail", and "Password". Each label is followed by a rectangular input field. Below these is a dropdown menu labeled "Select Department" with a small downward arrow icon. The dropdown menu contains the following options: CS, ME, EEE, IE, MATH, MBG, and ECON. At the bottom left of the form is a small checkbox followed by the text "I accept the terms of use". At the bottom center is a large red button with the white text "Sign up".

Figure 5.6.2.1: Sign Up Page For Students

In Figure 5.6.2.1, the student is asked to enter her name, surname, student ID, E-mail, password and department. After clicking on the "Sign up" button, an account is created using that info and the student will see the main page.

5.6.2.2 Sign Up Page For Instructor and Grader



The image shows a sign-up form for an instructor or grader. At the top left is a back arrow icon. To its right, the word "SIGN UP" is written in large, bold, black capital letters. Below this, there are four input fields: "Name", "Surname", "E-mail", and "Password", each with a thin black border. Below these is a dropdown menu labeled "Select Department" containing the following options: CS, ME, EEE, IE, MATH, MBG, and ECON. Underneath the dropdown is a small checkbox labeled "I accept the terms of use". At the bottom is a red rectangular button with the white text "Sign up".

Figure 5.6.2.2: Sign Up Page For Instructor and Grader

In Figure 5.6.2.2, the instructor is asked to enter her name, surname, e-mail, password and department. After clicking on the "Sign up" button, an account is created using that info and the student will see the main page. The only difference from the "Sign up Page for Students" page is that no student ID information is received from the user.

5.6.3 Login Page

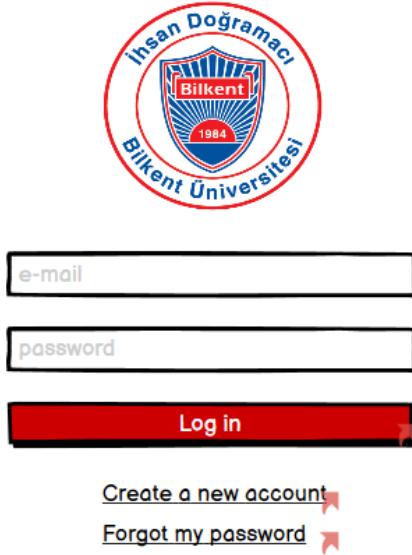
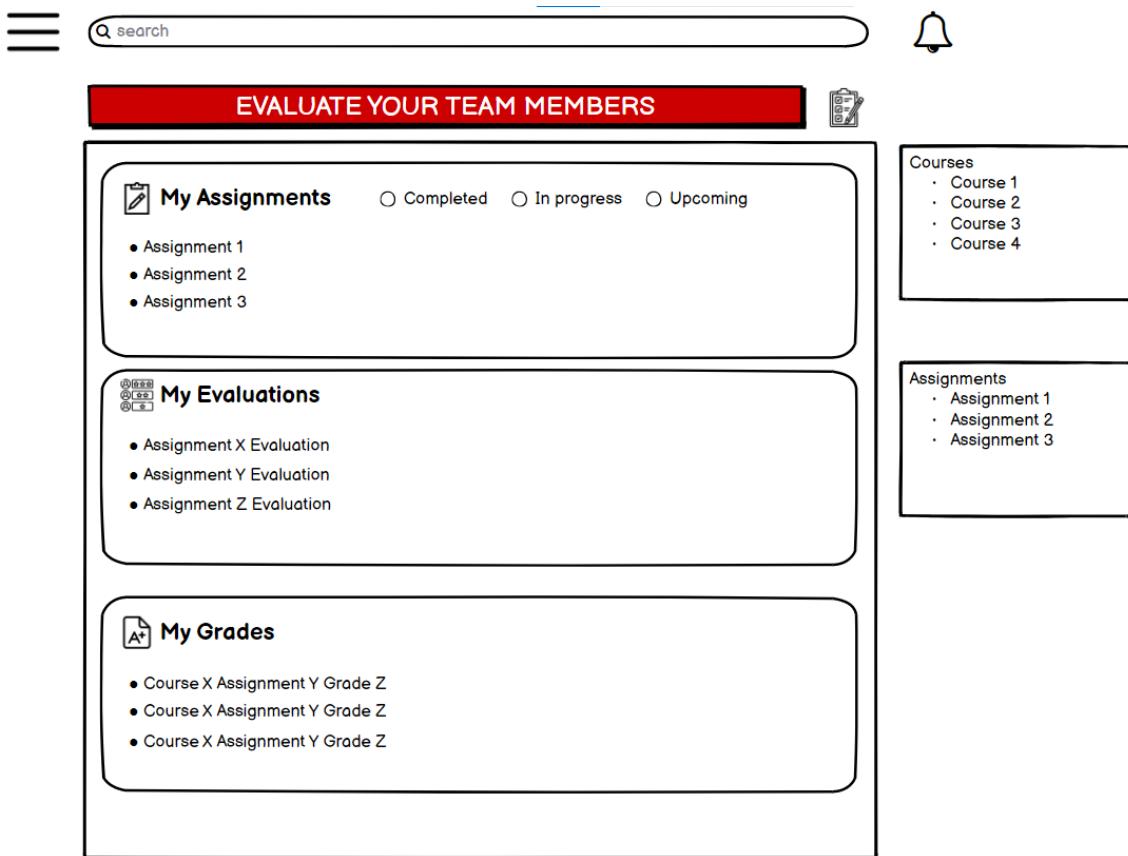


Figure 5.6.3: Login Page

In Figure 5.6.3, users log in to the application by entering their email and password. They can also choose the option of creating a new account or changing the passwords of their existing accounts in case they forgot it.

5.6.4 Main Page

5.6.4.1 Main Page For Students



The figure shows a wireframe of a user interface for students. At the top left is a menu icon (three horizontal lines). To its right is a search bar with a magnifying glass icon and the word "search". Further right is a bell icon. Below the search bar is a red button with the text "EVALUATE YOUR TEAM MEMBERS" in white. To the right of the button is a clipboard icon. The main content area contains three rounded rectangular boxes: "My Assignments" (with a clipboard icon), "My Evaluations" (with a document icon), and "My Grades" (with a grade icon). Each box lists items with bullet points. To the right of these boxes are two additional boxes: one for "Courses" listing four courses and one for "Assignments" listing three assignments.

- My Assignments**
 - Assignment 1
 - Assignment 2
 - Assignment 3
- Completed** **In progress** **Upcoming**
- My Evaluations**
 - Assignment X Evaluation
 - Assignment Y Evaluation
 - Assignment Z Evaluation
- My Grades**
 - Course X Assignment Y Grade Z
 - Course X Assignment Y Grade Z
 - Course X Assignment Y Grade Z

Courses <ul style="list-style-type: none">• Course 1• Course 2• Course 3• Course 4	Assignments <ul style="list-style-type: none">• Assignment 1• Assignment 2• Assignment 3
--	---

Figure 5.6.4.1: Main Page For Students

In Figure 5.6.4.1, users can evaluate the group members by clicking on the "Evaluate your team members" button when it is time to evaluate the assignments or the group near the end of the project. In addition, users can view their courses and tasks by clicking on the courses and assignments labels, view their assignments, evaluations and grades by clicking the relevant labels and finally click on the drop menu icon in the upper left corner to view options menu.

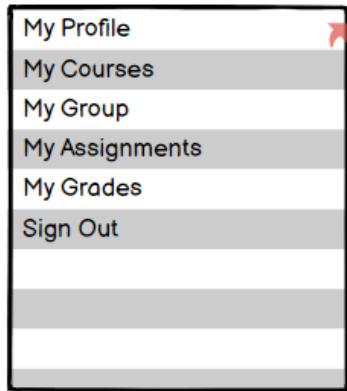


Figure 5.6.4.1.1: Drop Menu For Students

In Figure 5.6.4.1.1, the user can go to the related pages by pressing the my profile, my courses, my groups, my assignments and my grades tags. She can also sign out from the application.

5.6.4.2 Main Page For Instructor and Grader



Figure 5.6.4.2: Main Page For Instructor and Grader

Figure 5.6.4.2 is the home page for instructors and teaching assistants. There are 2 differences between this page and the "Main Page for Students" page. There is no "Evaluate your Team Member" button on this page and no "My Grades" label. Also, the drop menus are different for each user type.

5.6.4.2.1 Drop Menu For Instructor and Grader

In Figure 5.6.4.2.1, the user can go to the related pages by pressing the profile, courses and assignments tags. She can also sign out from the application.

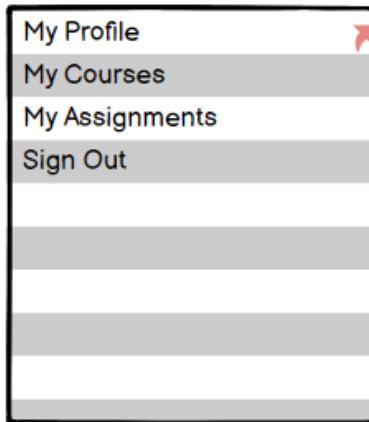


Figure 5.6.4.2.1: Drop Menu For Instructor and Grader

5.6.5 Profile Pages

5.6.5.1 Profile Page For User

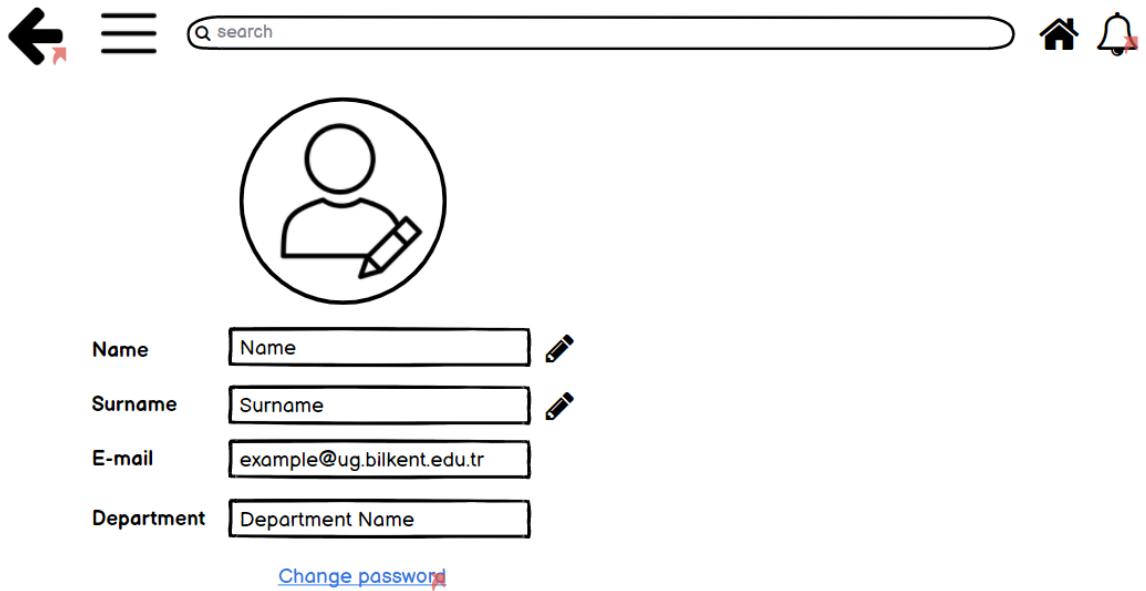


Figure 5.6.5.1: Profile Page For User

In Figure 5.6.5.1, users can view their profile information and edit them.

5.6.5.2 Profile Page For Other User

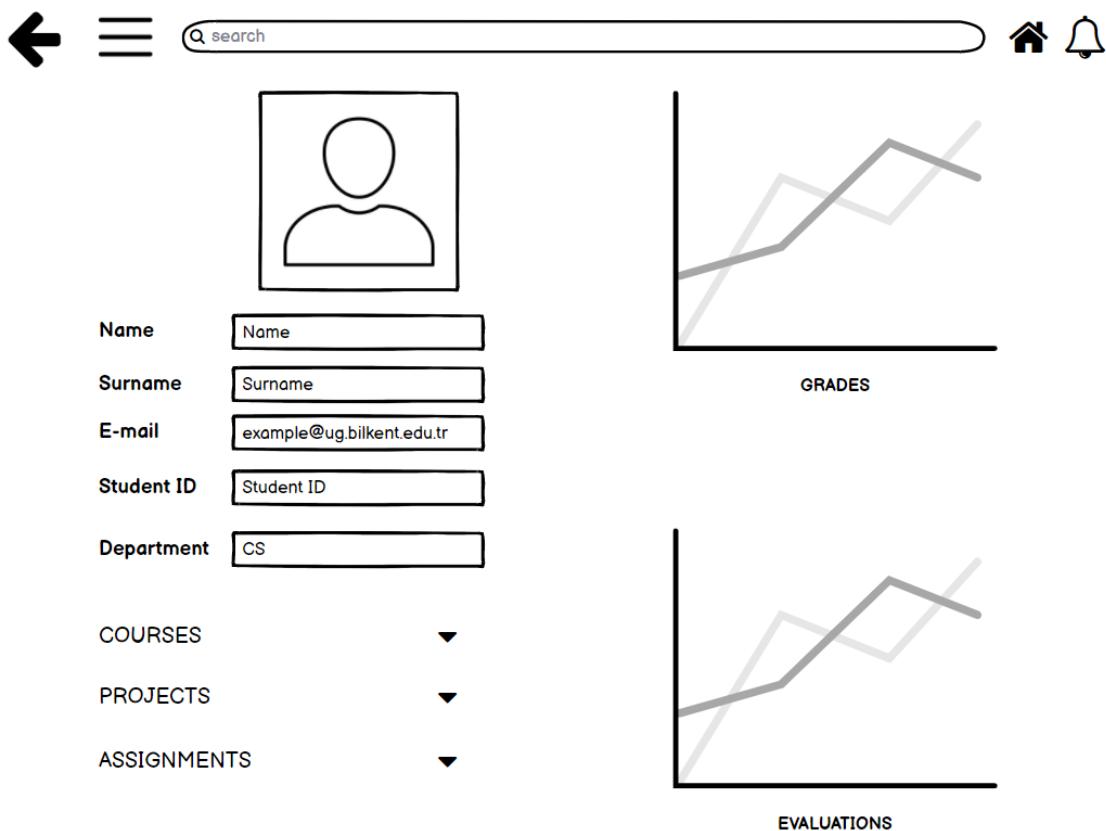


Figure 5.6.5.2 Profile Page For Other User

In Figure 5.6.5.2, users can view other users' profile information, courses, projects, assignments, grades, and evaluations, although the last two elements are optional and enabled by the instructor. Also, users can send a message to other users by clicking on the "Send Message" label. To access this page, users should click on person names under the "Members" tab on the group pages.

5.6.6 Courses Pages

5.6.6.1 Courses Page For Students

The screenshot shows a user interface for a student's course list. At the top, there is a navigation bar with a back arrow, a menu icon, a search bar containing the text "search", and icons for home and notifications. Below the navigation is a header with the word "COURSES" and a red "Enroll in a Course" button. Underneath the header are four rectangular boxes, each labeled with a course name: "COURSE 1", "COURSE 2", "COURSE 3", and "COURSE 4". Each box has a small red arrow icon at its bottom right corner.

Figure 5.6.6.1 Courses Page For Students

In Figure 5.6.6.1, students can view the courses they have enrolled in and go to the relevant course's page by clicking on its label. Also they can go to the relevant page to enroll in a new course by clicking on the "Enroll in a Course" button.

5.6.6.2 Courses Page For Instructor

The screenshot shows a user interface for an instructor's course list. At the top, there is a navigation bar with a back arrow, a menu icon, a search bar containing the text "search", and icons for home and notifications. Below the navigation is a header with the word "COURSES" and a red "Create a New Course" button. Underneath the header are four rectangular boxes, each labeled with a course name: "COURSE 1", "COURSE 2", "COURSE 3", and "COURSE 4". Each box has a small red arrow icon at its bottom right corner.

Figure 5.6.6.2 Courses Page For Instructor

In Figure 5.6.6.2, instructors can view the courses they have and go to the relevant course's page by clicking on its label. Also, they can go to the relevant page to create a course by clicking on the "Create a New Course" button.

5.6.6.3 Courses Page For Grader

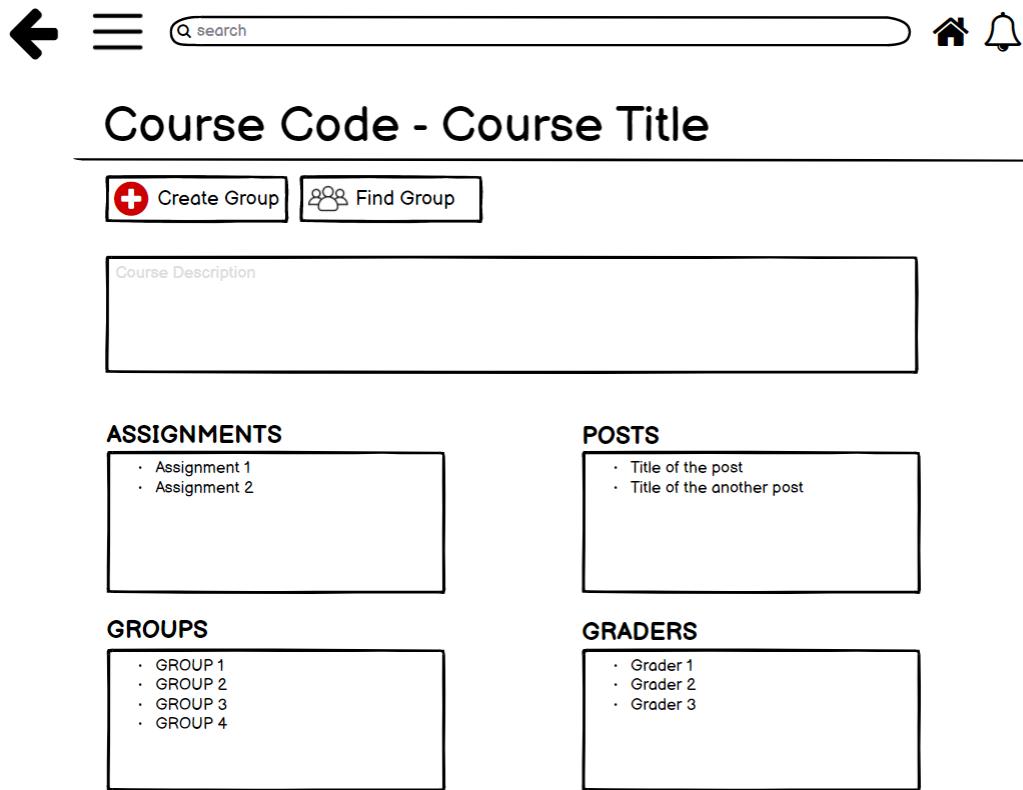
The screenshot shows a web-based application interface for a grader. At the top left is a back arrow icon. Next to it are three horizontal lines representing a menu. To the right of the menu is a search bar with the placeholder text "search". Further to the right are icons for a house and a bell, with a small red notification badge on the bell icon. Below this header, the word "COURSES" is centered in a large, bold, black font. A horizontal line separates the header from the main content area. The main content area contains four rectangular boxes, each labeled with a course name: "COURSE 1", "COURSE 2", "COURSE 3", and "COURSE 4". Each box has a small red circular icon with a white arrow pointing upwards and to the right in the bottom right corner.

Figure 5.6.6.3 Courses Page For Grader

In Figure 5.6.6.3, a grader can view the courses they grade. Also, they can go to the relevant course page by clicking on its label.

5.6.7 Detailed Course Pages

5.6.7.1 Detailed Course Page For Student



The figure shows a detailed course page for a student. At the top, there is a navigation bar with a back arrow, a menu icon, a search bar containing the text "search", and two icons: a house and a bell. Below the navigation bar, the course title "Course Code - Course Title" is displayed. Underneath the title, there are two buttons: "Create Group" (with a plus sign icon) and "Find Group" (with a group icon). A large empty box labeled "Course Description" follows. Below this, there are four sections: "ASSIGNMENTS" (containing "Assignment 1" and "Assignment 2"), "POSTS" (containing "Title of the post" and "Title of the another post"), "GROUPS" (containing "GROUP 1", "GROUP 2", "GROUP 3", and "GROUP 4"), and "GRADERS" (containing "Grader 1", "Grader 2", and "Grader 3").

Figure 5.6.7.1 Detailed Course Page For Student

In Figure 5.6.7.1, a student can view the course's description, assignments, posts, groups and graders. If the student clicks on assignments, posts or groups labels, she is directed to the pages where she will see the clicked label's page. Also, if she is not in a group, she can create a new group by clicking on the "Create Group" button or find a group by clicking on the "Find Group" button.

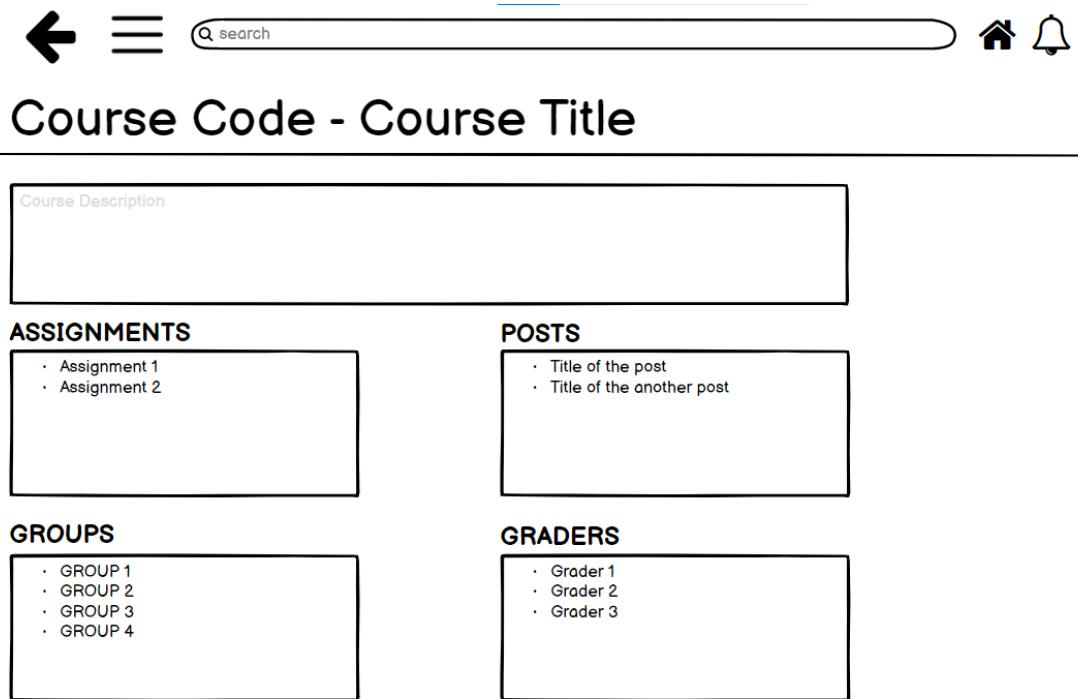
5.6.7.2 Detailed Course Page For Instructor

The screenshot shows a web-based course management system. At the top, there is a header bar with a back arrow, a menu icon, a search bar containing the placeholder "search", and icons for a house and a bell. Below the header, the page title is "Course Code - Course Title". Underneath the title are two buttons: "Create Project" and "Create Assignment", both featuring a red circle with a white plus sign. A large rectangular box labeled "Course Description" is present, with a small pencil icon in the bottom right corner indicating it is editable. Below this, there are four sections: "ASSIGNMENTS" (listing "Assignment 1" and "Assignment 2"), "POSTS" (listing "Title of the post" and "Title of the another post"), "GROUPS" (listing "GROUP 1", "GROUP 2", "GROUP 3", and "GROUP 4"), and "GRADERS" (listing "Grader 1", "Grader 2", and "Grader 3").

Figure 5.6.7.2 Detailed Course Page For Instructor

In Figure 5.6.7.2, an instructor can edit the course's description and see the assignments, posts, groups and graders. If the instructor clicks on assignments, posts or groups labels, she is directed to the pages where she will see the clicked label's page. Also, the instructor can create projects and assignments for the course by clicking on the "Create Project" and "Create Assignment" buttons.

5.6.7.3 Detailed Course Page For Grader

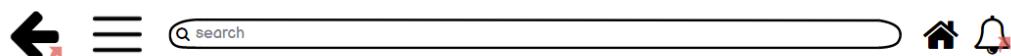


The image shows a mobile-style user interface for a course page. At the top, there is a navigation bar with a back arrow, a menu icon, a search bar containing the word "search", and icons for home and notifications. Below the navigation bar, the course title "Course Code - Course Title" is displayed. The main content area is organized into four sections: "Course Description" (empty box), "ASSIGNMENTS" (list: Assignment 1, Assignment 2), "POSTS" (list: Title of the post, Title of the another post), "GROUPS" (list: GROUP 1, GROUP 2, GROUP 3, GROUP 4), and "GRADERS" (list: Grader 1, Grader 2, Grader 3). Each section is contained within its own rectangular box.

Figure 5.6.7.3 Detailed Course Page For Grader

In Figure 5.6.7.3, a teaching assistant can see the course's description, assignments, posts, groups and graders. If the teaching assistant clicks on assignments, posts or groups labels, she is directed to the pages where she will see the clicked label's page.

5.6.8 Assignments Page



Course Code - Course Title - Assignments

Assignment 1

Assignment 2

Assignment 3

Assignment 4

- █ Completed
- █ In progress
- █ Upcoming

Figure 5.6.8 Assignments Page

Figure 5.6.8 is the same for all users. By clicking on the assignments' labels, users can access the page where they can view the details of the assignments. The green color indicates that the assignment is completed, the yellow color indicates that it is in progress, and the red color indicates that it is an upcoming one.

5.6.9 Detailed Assignment Pages

5.6.9.1 Detailed Not Completed Assignment Page For Students

The screenshot shows a web-based assignment submission interface. At the top, there are navigation icons: a left arrow, a menu icon, a search bar with the placeholder 'search', a home icon, and a bell icon with a red notification badge. Below the header, the title 'Assignment Title' is displayed. The main content area contains three sections: 'Assignment Description' with a large text box containing Latin placeholder text; 'Assignment Deadline' with a date field showing '16.03.2021'; and 'Submit Assignment' with a large input field featuring a hand cursor icon and a plus sign. At the bottom, there are two buttons: a red 'SUBMIT' button and a grey 'EDIT SUBMISSION' button.

Figure 5.6.9.1 Detailed Not Completed Assignment Page For Students

In Figure 5.6.9.1, students can view the description and deadline of the assignment. They can also submit the assignment by dragging and dropping the assignment file or by uploading the file and editing the assignment they submitted.

5.6.9.1.1 Detailed Completed Assignment Page For Students

The screenshot shows a user interface for a completed assignment. At the top, there is a navigation bar with a back arrow, a menu icon, a search bar containing the placeholder "search", and icons for a house and a bell. Below the navigation bar, the title "Assignment Title" is displayed in a large, bold font. To the right of the title is a red button labeled "Evaluate Your Team Members". A horizontal line separates this section from the main content area. In the main content area, there are two sections: "Assignment Description" and "Assignment Deadline". The "Assignment Description" section contains a block of Latin text: "Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt." The "Assignment Deadline" section contains the date "16.03.2021". Below these sections, there is a link labeled "Submission File" with a document icon, and another link labeled "Click to see the responses to the reviews" with a blue underline.

Figure 5.6.9.1.1 Detailed Completed Assignment Page For Students

In Figure 5.6.9.1.1, students can view information on completed assignments. Also, if they are viewing their group's assignment page, they can go to the page where they can evaluate their group members by clicking the "Evaluate Your Team Members" button on the top right.

5.6.9.2 Detailed Assignment Page For Instructors

The screenshot shows a detailed assignment page for instructors. At the top, there are navigation icons: a left arrow, a menu icon, a search bar with a magnifying glass icon, a home icon, and a bell icon with a red notification badge. Below the header, the title "Assignment Title" is displayed, followed by a "Grade Assignment" button. The main content area includes fields for "Assignment Description" (containing placeholder Latin text), "Assignment Deadline" (set to "16.03.2021"), a dropdown menu for "Select Progress" (listing "Completed", "In Progress", and "Upcoming"), and a toggle switch labeled "Evaluation time". A large red "Save" button is located at the bottom.

Assignment Description	Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt.
Assignment Deadline	16.03.2021
Select Progress	Completed In Progress Upcoming
Evaluation time	(Toggle switch)
Save	

Figure 5.6.9.2 Detailed Assignment Page For Instructors

In Figure 5.6.9.2, instructors can edit the name, description and deadline of the assignment. They can also select the progress for the assignment and determine if it is time to evaluate or not. Finally, if the assignment's deadline has passed, they can click on the "Grade Assignment" button to reach the page where they will grade the assignment.

5.6.9.3 Detailed Assignment Page For Grader

The screenshot shows a web-based assignment management system. At the top left is a back arrow icon. Next to it are three horizontal lines indicating a menu. A search bar with the placeholder "search" is positioned next. On the far right are icons for a house and a bell, with a small red square notification badge above the bell icon.

The main title "Assignment Title" is centered at the top. To its right is a red button labeled "Grade Assignment".

Below the title, there is a section titled "Assignment Description" which contains a large block of Latin placeholder text:

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt.

Further down, there is a row with two entries: "Assignment Deadline" followed by the date "16.03.2021".

Figure 5.6.9.3 Detailed Assignment Page For Grader

In Figure 5.6.9.3, teaching assistants can see the name, description and deadline of the assignment. They can also reach the page where they will grade the assignment by clicking the "Grade Assignment" button.

5.6.10 Groups Pages

5.6.10.1 Groups Page For Students

The screenshot shows a user interface for managing groups. At the top, there is a navigation bar with a back arrow, a menu icon, a search bar containing the word "search", and icons for home and notifications. Below the navigation bar, the word "GROUPS" is displayed in large, bold capital letters. Underneath, there are four separate sections, each labeled with a group name and a small user icon:

- GROUP 1**: Contains three items: "Member 1 - Member 2", "Member 3 - Member 4", and "Member 5".
- GROUP 2**: Contains three items: "Member 1 - Member 2", "Member 3 - Member 4", and "Member 5".
- GROUP 3**: Contains three items: "Member 1 - Member 2", "Member 3 - Member 4", and "Member 5".
- GROUP 4**: Contains three items: "Member 1 - Member 2", "Member 3 - Member 4", and "Member 5".

Figure 5.6.10.1 Detailed Assignment Pages

In Figure 5.6.10.1, students can see the groups and members of the relevant course. If they click on their own group's label, they are directed to their own group's page, if they click on another group's label, they are directed to the pages of the said group.

5.6.10.2 Groups Page For Instructor And Graders

The screenshot shows a web-based application interface for managing groups. At the top left is a back arrow, followed by a menu icon (three horizontal lines) and a search bar containing the word "search". To the right are icons for home and notifications. Below the header, the word "GROUPS" is centered in a large, bold, black font.

The main content area displays four groups, each with a title and a list of members:

- GROUP 1**: Contains Member 1 - Member 2, Member 3 - Member 4, and Member 5. Below it is a red "Grade" button.
- GROUP 2**: Contains Member 1 - Member 2, Member 3 - Member 4, and Member 5. Below it is a red "Grade" button.
- GROUP 3**: Contains Member 1 - Member 2, Member 3 - Member 4, and Member 5. Below it is a red "Grade" button.
- GROUP 4**: Contains Member 1 - Member 2, Member 3 - Member 4, and Member 5. Below it is a red "Grade" button.

Figure 5.6.10.2 Groups Page For Instructor And Graders

In Figure 5.6.10.2, instructors and teaching assistants can see the groups and members of the relevant course. If they click on a group's label, they are directed to the clicked group's page. Also, they can reach the page where they will choose all assignments of a group by clicking on the "Grade" button.

5.6.11 Groups Assignments Page For Students Instructor and Graders

The screenshot shows a web page titled "GROUP 1 ASSIGNMENTS". At the top left are navigation icons: a back arrow, a menu icon, and a search bar with placeholder text "search". To the right are a home icon and a bell icon with a red notification dot. Below the title is a horizontal line. Underneath the line, there are four identical rectangular boxes, each labeled "Assignment" and featuring a small red arrow icon at the bottom right corner.

Figure 5.6.11 Groups Assignments Page For Students Instructor and Graders

In Figure 5.6.11, students, instructors, and teaching assistants can see the assignments of the clicked group. Students can reach this page by clicking on the "Evaluate" button in the "Detailed Other Groups Page". If they click on assignment labels, they are directed to the clicked evaluation page. Also, instructors and teaching assistants can reach this page by clicking the "Grade" button in the "Groups Page for Instructors and TA". If they click on an assignment's label, they are directed to the clicked evaluation's page.

5.6.12 Grading Page For Instructors and Graders

5.6.12.1 Grade Group Page For Instructors and Graders

The screenshot shows a web-based grading interface. At the top left are navigation icons: a back arrow, a menu icon, and a search bar with the placeholder "search". To the right are a house icon, a bell icon, and a blue "Grade Individually" button. Below these, the title "ASSIGNMENT 1" is centered. The main content area contains three assignment entries, each with a label, a score input field, and a notes section. The first entry is labeled "Criteria 1" with a "/10" input field. The second entry is labeled "Criteria 2" with a "/10" input field. The third entry is labeled "Criteria 2" with a "/10" input field. Each entry has a "Notes" section below it. At the bottom is a large red "Submit" button.

Figure 5.6.12.1 Grade Group Page For Instructors and Graders

In Figure 5.6.12.1, instructors and teaching assistants can grade and give feedback for one of the assignments of the group. They can reach this page by clicking one of the assignments' labels in the group assignments page. In addition, they can evaluate the members of this group individually by clicking on the "Grade Individually" button.

5.6.12.2 Grade Individually Page For Instructor and Graders

The screenshot shows a web-based assignment grading interface. At the top left are navigation icons: a back arrow, a menu icon, and a search bar with the placeholder 'search'. To the right are a home icon and a bell icon. Below the header is the title 'ASSIGNMENT 1' in large bold letters, with a 'Grade as a Group' button to its right. A horizontal line separates the header from the main content.

Member 1

Criteria 1 /10
Criteria 2 /10
Criteria 3 /10

Notes

Member 2

Criteria 1 /10
Criteria 2 /10
Criteria 3 /10

Notes

Submit

Figure 5.6.12.2 Grade Individually Page For Instructor and Graders

In Figure 5.6.12.2, instructors and teaching assistants can grade and give feedback for an assignment for a student. In addition, they can evaluate the whole group by clicking on the "Grade as a Group" button.

5.6.13 Detailed Group Page

5.6.13.1 Detailed Your Group Page For Students

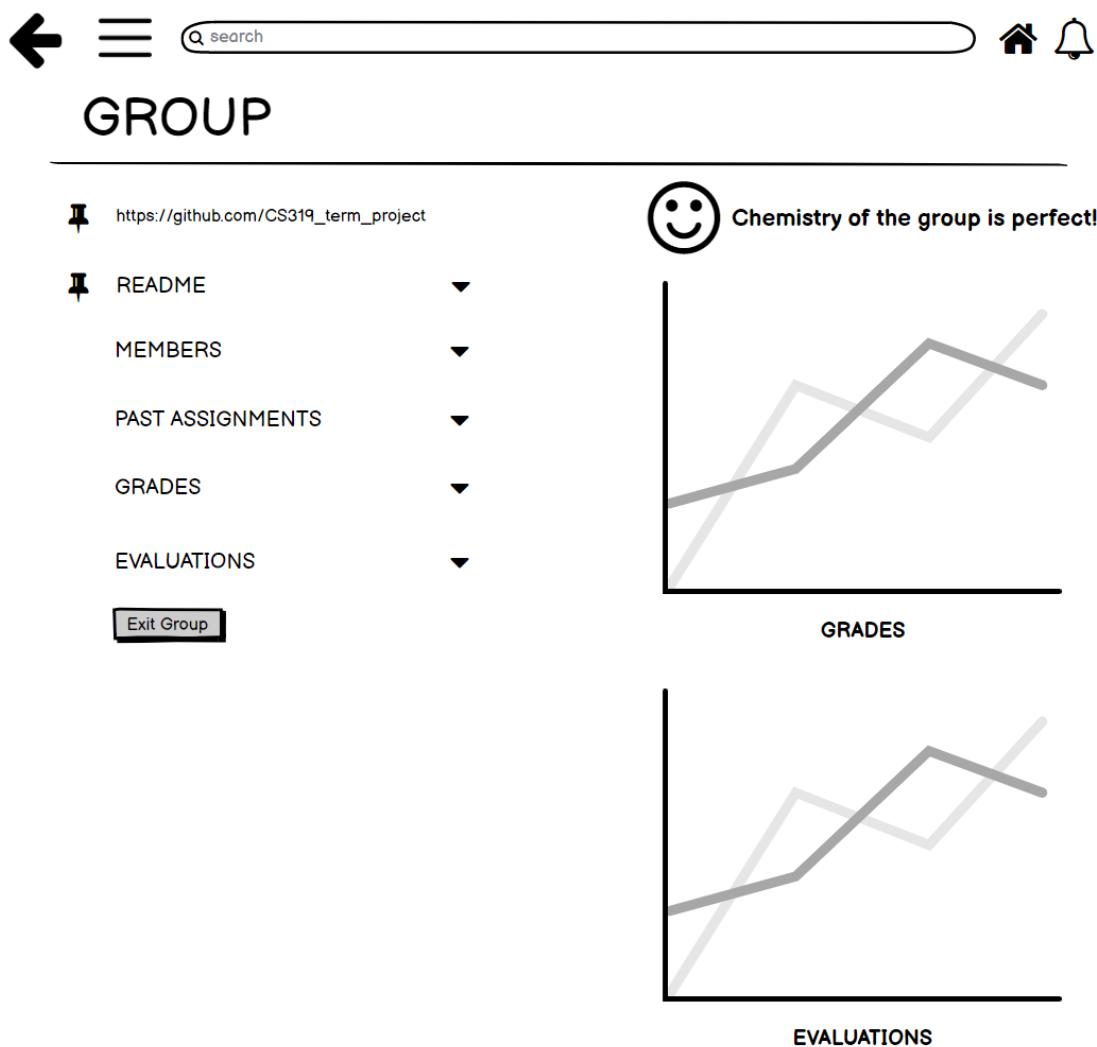


Figure 5.6.13.1 Detailed Your Group Page For Students

In Figure 5.6.13.1, students can see their group's members, assignments, grades, evaluations, chemistry and related graphs. Also, if the group formation phase is not over, they can exit the group by clicking the "Exit Group" button.

5.6.13.2 Detailed Other Groups Page For Students

The screenshot shows a web-based application interface for a group named 'GROUP 1'. At the top left is a back arrow icon. Next to it are three horizontal bars representing a menu. To the right of the menu is a search bar with the placeholder 'search'. Further right are icons for a home page and a notification bell, which has a red badge indicating one new notification.

The main content area has a title 'GROUP 1' in large bold letters. Below the title is a red button labeled 'Evaluate'. On the left side, there is a sidebar with several items: a link to 'https://github.com/CS319_term_project', a 'README' section, a 'MEMBERS' section, a 'PAST ASSIGNMENTS' section, and an 'EVALUATIONS' section. Each item in the sidebar has a small downward arrow icon to its right.

On the right side of the page, there is a graphic element consisting of a smiley face icon followed by the text 'Chemistry of the group is perfect!'. Below this is a line graph with the word 'EVALUATIONS' written horizontally below the x-axis. The graph features a light gray line with a darker gray shaded area underneath, showing an upward trend from left to right.

Figure 5.6.13.2 Detailed Other Groups Page For Students

In Figure 5.6.13.2, students can see other groups' members, assignments, chemistry and related graphics. They can also evaluate other groups by clicking the "Evaluate" button.

5.6.13.3 Detailed Other Groups Page For Instructor and Grader

The screenshot shows a web-based application interface for managing a group. At the top left is a back arrow icon. Next to it are three horizontal bars representing a menu. To the right is a search bar with a magnifying glass icon and the word "search". Further right are icons for a home page and a bell with a red notification dot. The main title "GROUP 1" is centered above a horizontal line. To the right of the title is a red "Grade" button. Below the title, there is a list of group components with dropdown arrows: "https://github.com/CS319_term_project", "README", "MEMBERS", "PAST ASSIGNMENTS", "GRADES", and "EVALUATIONS". To the right of this list are two line graphs. The first graph, titled "EVALUATIONS", shows a light gray line with several sharp peaks and troughs. The second graph, titled "GRADES", shows a similar light gray line with more rounded peaks and troughs. Above the "EVALUATIONS" graph is a smiley face icon and the text "Chemistry of the group is perfect!".

Figure 5.6.13.3 Detailed Other Groups Page For Instructor and Grader

In Figure 5.6.13.3, instructors and teaching assistants can see the group's members, assignments, grades, evaluations, chemistry and related graphs. They can also grade the group by clicking the “Grade” button.

5.6.14 Evaluation Pages

5.6.14.1 Evaluate Your Group Member Page

The screenshot shows a web-based application interface for evaluating group members. At the top, there are navigation icons: a left arrow, a menu icon, a search bar with placeholder text 'search', a home icon, and a bell icon. Below the header, the title 'EVALUATE YOUR GROUP MEMBERS' is centered.

Member 1

Criteria 1

1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	----

Criteria 2

1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	----

Criteria 3

1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	----

Notes

EVALUATE

Member 2

Criteria 1

1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	----

Criteria 2

1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	----

Criteria 3

1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	----

Notes

EVALUATE

Figure 5.6.14.1 Evaluate Group Members Page

In Figure 5.6.14.1, students can evaluate their group members according to given criteria. To go to this page, click on the "Evaluate Your Team Members" button on the "Detailed Assignment Page for Students" page or select the assignment after clicking the "Evaluate Your Team Members" button that will appear when the time comes on the "Main Page for Students" page.

5.6.14.2 Evaluate Other Groups Page

The screenshot shows a web-based application for evaluating assignments. At the top, there is a navigation bar with a back arrow, a menu icon, a search bar containing the word 'search', and icons for home and notifications. Below the navigation is a section titled 'Assignment Title' with a 'See responses to the reviews' button. A large empty box labeled 'Assignment Description' follows. Underneath, there is a section for 'Submission File'. Below this are six rows, each labeled 'Criteria' (1 through 6) and featuring a circular rating scale from 1 to 10. The scales for Criteria 1, 2, 3, 4, and 5 are colored orange, while the one for Criterion 6 is green. A notes section with a 'Notes' placeholder is present, and at the bottom is a prominent red 'EVALUATE' button.

Figure 5.6.14.2 Evaluate Other Groups Page

In Figure 5.6.14.2, students can access other groups' assignment files and evaluate groups according to given criteria. They can also see the responses posted by the group to the reviews and respond to the responses. Students can access this page by clicking on the "Evaluate" button on the "Detailed Other Groups Page for Students" page, and selecting the assignment they want to evaluate from the "Group Assignments Page for Students, Instructors and TA" page.

5.6.15 Create Course Page For Instructors

Course Name

Course Code

Course Description

Entry Code

Select Grader
Grader 1
Grader 2
Grader 3

CREATE COURSE

Figure 5.6.15 Create Course Page For Instructors

In Figure 5.6.15, instructors can create a course by entering the required information for the course. "Entry Code" is a unique code that will be created for students to enroll in the course.

5.6.16 Create Project Page For Instructor

The screenshot shows a web-based application interface for creating a project. At the top, there is a navigation bar with a back arrow, a menu icon, a search bar containing the placeholder 'search', and icons for home and notifications. Below the navigation is a title 'CREATE PROJECT' and a red button labeled 'Assign Students to Groups'. The main form consists of several input fields and dropdown menus:

- Select Course:** A dropdown menu showing options: CS 319, CS 315, CS 353, CS 342.
- Project Name:** An empty text input field.
- Project Description:** A large empty text area.
- Project Language:** A dropdown menu showing options: C++, Java, C#, Python.
- Project Tools:** An empty text input field.
- Group Size:** A dropdown menu showing options: 2-3, 3-4, 4-5, 5-6.

Below the form is a checkbox labeled 'Random Groups' and a red 'Create Project' button.

Figure 5.6.16 Create Project Page For Instructor

In Figure 5.6.16, instructors can create a project by entering the required information for the project. They can also assign students to groups either randomly or manually. If the "Random Groups" checkbox is checked, the groups are formed randomly. If the "Assign Students to Groups" button is clicked, the instructor is directed to the page where the groups can be formed manually.

5.6.17 Assign Student to Groups Page

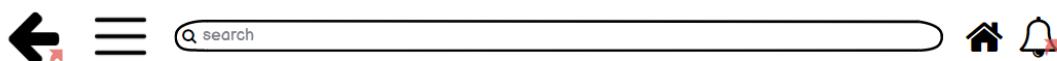
The screenshot shows a user interface for managing student groups. At the top, there is a navigation bar with a back arrow, a menu icon, a search bar containing the text 'search', a home icon, and a bell icon with a red notification dot. Below the navigation bar are two buttons: 'Add Group' (white background with a red plus sign) and 'Random Groups' (red background). The main area is divided into four sections labeled 'Group 1', 'Group 2', 'Group 3', and 'Group 4', each containing a square icon with a hand and four small dots. To the right of these groups is a 'Student List' table with 15 rows, each representing a student numbered from 1 to 15. The rows alternate in color between white and light gray.

Student List
Student 1
Student 2
Student 3
Student 4
Student 5
Student 6
Student 7
Student 8
Student 9
Student 10
Student 11
Student 12
Student 13
Student 14
Student 15

Figure 5.6.17 Assign Student to Groups Page

In Figure 5.6.17, instructors can create groups for the project, delete groups, assign students to groups with drag and drop, or create groups randomly.

5.6.18 Grades Page



Course Code - GRADES

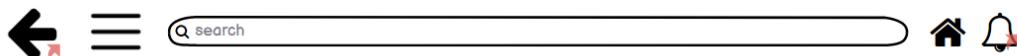
Select Course ▼

Name	Grade	Average
Assignment 1	6/10	5.4/10
Assignment 2	10/10	8/10
Assignment 3	3/10	4/10
Assignment 3	7.5/10	6.8/10

Figure 5.6.18 Grades Page

In Figure 5.6.18, students can see their grades and average grades for each course they are enrolled in. Students can click on the "Grades" tag from the drop menu on their homepage to access this page.

5.6.19 Notifications Page



NOTIFICATIONS

The first line of Lorem Ipsum, "Lorem ipsum dolor sit amet..", comes from a line in section 1:10.32

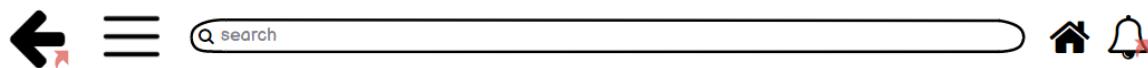
Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur? Quis autem vel eum iure reprehenderit qui in ea voluptate velit esse quam nihil molestiae consequatur, vel illum qui dolorem eum fugiat quo voluptas nulla pariatur?"

Itaque earum rerum hic tenetur a sapiente delectus, ut aut reiciendis voluptatibus maiores alias consequatur aut perferendis doloribus asperiores repellat."

Figure 5.6.19 Notifications Page

Users can view their notifications in Figure 5.6.19. If they click on the notifications, they reach the relevant page.

5.6.20 Enroll in a Course Page For Students



Entry Code

Enroll

Figure 5.6.20 Enroll in a Course Page For Students

In Figure 5.6.20, students enroll in the course by entering the unique course code given to them by the instructors. To access this page, users should click on the "Enroll in a Course" button on the "Courses Page for Students" page.

5.6.21 Create Assignment Page For Instructors

A screenshot of a web page with a header containing a back arrow, a menu icon, a search bar, and a house/bell icon. The main title is "CREATE ASSIGNMENT". Below the title, there are several input fields: "Select Course" dropdown (options: CS 319, CS 315, CS 353, CS 342), "Select Project" dropdown (options: CS 319 Term Project, CS 353 Term Project), "Assignment Name" input field, "Assignment Description" text area, "Choose a Deadline" input field with a calendar icon, and a red "Create Assignment" button.

Figure 5.6.21 Create Assignment Page For Instructors

In Figure 5.6.21, instructors can create a new assignment after entering the required information for the assignment.

5.6.22 Create Group Page

The screenshot shows a mobile-style interface for creating a group. At the top, there is a navigation bar with a back arrow, a menu icon, a search bar containing the placeholder 'search', and icons for home and notifications. Below the navigation is the title 'CREATE GROUP'. On the left, a dropdown menu titled 'Select Course' lists three options: CS 319, CS 342, and CS 315. To the right of the course list are six student profile icons arranged in two rows of three. Each profile includes a 'Student Name' label and a 'Select' checkbox. At the bottom right is a red button labeled 'Send Request'.

Student Name	Select
□ Select	
□ Select	
□ Select	
□ Select	
□ Select	
□ Select	
□ Select	
□ Select	
□ Select	
□ Select	
□ Select	
□ Select	

Send Request

Figure 5.6.22 Create Group Page

In Figure 5.6.22, students can send a request to other students to create a new group if the group formation process is ongoing. To access this page, students should click on the "Create Group" button on the "Detailed Course Page for Students" page.

5.6.23 Find Group Page For Students

The screenshot shows a web page titled "GROUPS". There are four groups listed:

- GROUP 1**: Contains "Member 1 - Member 2" and "Member 3 - Member 4".
- GROUP 2**: Contains "Member 1 - Member 2".
- GROUP 3**: Contains "Member 1 - Member 2" and "Member 3 -".
- GROUP 4**: Contains "Member 1 - Member 2", "Member 3 - Member 4", and "Member 5".

Each group box has a red arrow pointing to each member name.

Figure 5.6.23 Find Group Page For Students

If students click on the "Find Group" button on the "Detailed Course Page for Students" page, they will reach this page. In Figure 5.6.23, they can see the groups in the formation process. If they click on a group's label, they will be redirected to another page where they can send a request to join the group.

5.6.24 Send Request To Join a Group For Students

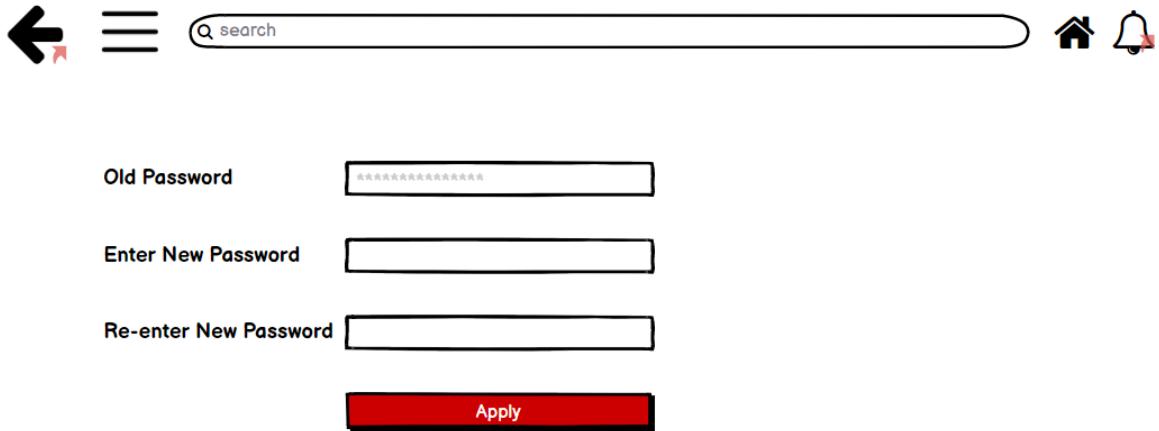
The screenshot shows a web page for "GROUP 1". It includes:

- A "Send Join Request" button.
- A link to "https://github.com/CS319_term_project".
- A "README" section with a dropdown arrow.
- A "MEMBERS" section with a dropdown arrow.

Figure 5.6.24 Send Request to Join a Group For Students

In Figure 5.6.24, students can see the members and other information of the group that is in the formation process and send a request to join the group by clicking on the "Send Join Request" button.

5.6.25 Change Password Page



The screenshot shows a mobile-style interface for changing a password. At the top, there is a back arrow icon, a menu icon (three horizontal lines), a search bar with the placeholder "search", and icons for home and notifications. Below the header, there are three input fields: "Old Password", "Enter New Password", and "Re-enter New Password". Each field has a placeholder consisting of ten question marks. At the bottom is a red "Apply" button.

Old Password	<input type="text"/>
Enter New Password	<input type="text"/>
Re-enter New Password	<input type="text"/>
<input type="button" value="Apply"/>	

Figure 5.6.25 Change password page

In Figure 5.6.25, users can change their passwords. To access this page, the user should go to their profile and click on the "Change Password" button.

5.6.26 Review Responses to the Reviews Page

The figure consists of two vertically stacked screenshots of a digital platform's review section. Both screenshots feature a header with a back arrow, a menu icon, a search bar, and navigation icons for home and notifications.

Screenshot 1 (Top):

- Section Header:** Review 1 - Name Surname
- Text:** Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo.
- Section Header:** Response 1 - Name Surname
- Text:** At vero eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesentium voluptatum deleniti atque corrupti quos dolores et quas molestias excepturi sint occaecati cupiditate non provident, similiq[ue] sunt in culpa qui officia deserunt mollitia animi, id est laborum et dolorum fuga.
- Section Header:** Response to Response - Name Surname
- Text:** At vero eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesentium voluptatum deleniti atque corrupti quos dolores et quas molestias excepturi sint occaecati cupiditate non provident, similiq[ue] sunt in culpa qui officia deserunt mollitia animi, id est laborum et dolorum fuga.
- Input Field:** Write a response
- Attachment Icon:** Paperclip icon
- Button:** Send

Screenshot 2 (Bottom):

- Section Header:** Review 2 - Name Surname
- Text:** Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo.
- Section Header:** Response 2 - Name Surname
- Text:** At vero eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesentium voluptatum deleniti atque corrupti quos dolores et quas molestias excepturi sint occaecati cupiditate non provident, similiq[ue] sunt in culpa qui officia deserunt mollitia animi, id est laborum et dolorum fuga.
- Input Field:** Write a response
- Attachment Icon:** Paperclip icon
- Button:** Send

Figure 5.6.25 Review Responses to the Reviews Page

In Figure 5.6.25, students can respond to reviews made to their own group or other groups. They can also discuss the topic via Slack by clicking the "Discuss in Slack" button. Also, students can add attachments to their responses.

6 Summary and Improvements

In our web application, PeeReview, we provide a service for students to be able to evaluate each other in group projects. The application has three types of users, student, grader, and instructor, each with different authority and access levels. Our system is composed of multiple key elements, this includes, the course, project, group, evaluation, and submission elements, each with its vital role in the system. The requirements of the applications are what is expected from such a service, for example, a course, project, and groups can be created, assignments can be submitted, graded, and evaluated. Along with other complementary features, like responding to reviews, analyzing and visualizing the evaluations, notify users, forming groups in multiple different ways and adding other services to the system, like Slack application and mail services. Uses cases, state machine, activity, sequence and class diagrams were made to represent and explain the system in the clearest way possible. Using these diagrams different levels of application can be seen and explained. We also designed a simple User Interface, that is shown in the report, to make reaching the different properties and components as easy as possible. Improvements include more defined and clearer state and activity diagrams as well as different structure for the classes. We also stressed some features like reviews and responses more after brainstorming activities. Additionally, a more focused application is designed now, with the removal of the formula and chat services, more energy and effort can be directed towards key features, enhancing the usability of the program. Finally, with realizing more possible features different key components and requirements were added like, the responses and follow-ups and resubmission features defining how other services like Slack and mail services will be used.