



Bilkent University

Department of Computer Engineering

CS319

Object-Oriented Software Engineering

Classroom Helper

Analysis Report

Tuna Dalbeler (21802539), Ezgi Saygılı (21802871), Funda Tan (21801861),
Selahattin Cem Öztürk (21802856), Onur Korkmaz(21802925)

Instructor: Eray Tüzün

Teaching Assistants: Elgun Jabrayilzade, Erdem Tuna

1.Introduction	2
2.Proposed System	2
2.1 Overview	2
2.2 Functional Requirements	4
2.3 Nonfunctional Requirements	6
2.4 Pseudo Requirements	7
3. System models	8
3.1 Use case model	8
3.2 Sequence Diagrams	19
Scenario 1: Login	19
Scenario 2: Join a Group	20
Scenario 3: Upload Assignment	21
Scenario 4: Tasks	22
Scenario 5: In Group Peer Review	23
3.3 Activity Diagrams	24
Activity Diagram 1: Login	24
Activity Diagram 2: My Group Page	25
Activity Diagram 3: All Groups Page (Students)	26
Activity Diagram 4: All Groups Page (Teachers)	27
3.4 Object and Class Model	28
3.5 User interface - navigational paths and screen mock-ups	31
3.5.1 Navigational Paths	31
3.5.2 Screen Mock-Ups	32
3.5.2.1 Login Page	32
3.5.2.2 Sign Up Page	32
3.5.2.3 My Courses Page	33
3.5.2.4 Course Chat Page	33
3.5.2.5 My Group Page	34
3.5.2.6 All Groups Page	34
3.5.2.7 Group Review Page	35
3.5.2.8 Grade Book page	35
3.5.2.9 Grade Book page continues (Instructor view)	36
3.5.2.10 People Page	36
3.5.2.11 Private Chat Page	37
3.5.2.12 Course Calendar Page	37
3.5.2.13 Course Related Info	38
3.5.2.14 Assignments Page (Instructor view)	38
3.5.2.15 Create Assignment Page	39
3.5.2.16 Show Assignment Page	39
3.5.2.17 Grade Assignment Page	40
3.5.2.18 Student Forum Page	40
3.5.2.19 Create Post Page	41
3.5.2.20 Show Post Page	41
4. Glossary & References	42

1.Introduction

Our project is intended to simplify the group formation and peer review processes in classes, which is called a Classroom Helper. With our application, finding groups and reviewing and grading their groups or the other groups in the classroom will be easier. Moreover, it will facilitate the communication between students, teachers and teaching assistants with the student forums and synchronous chat function including private, group and general messaging and information about teachers and teaching assistants. It facilitates in-group activities with an in-group work distribution tool and a deadline calendar. Also, it provides some features for instructors such as analytic tools. We will use Java to implement our project.

2.Proposed System

2.1 Overview

Group Formation:

Group formation consists of sending group requests to other groups and inviting other students when you have a group. Group formation includes the “bundle me with a friend” option. If the group is not full, students may send a request, otherwise, they cannot. Teachers may start or end the group formation process.

Peer Review and Peer Grading :

The peer-review feature enables students to review and grade their teammates and the other groups. Teachers and teaching assistants can see the reviews and grades of the groups.

Synchronous Chat:

The synchronous chat function enables synchronous communication between teachers, teaching assistants and students.

Messaging:

Messaging includes private messaging between members of the classroom and group messaging between the people in the group and general messaging between people in the course.

Student Forums:

Student forums allow students, teachers and teaching assistants to ask questions and post answers.

Tasks:

Students may create tasks or delete tasks using tasks functionality. With tasks, students can distribute the work among the group members.

Upload Assignment:

Students can upload assignments using the upload assignment tool. By doing this, other groups can see their assignments to review and grade.

Calendar:

The Calendar shows due dates and dates that may be important. Users can add events to the calendar or delete events from the calendar.

Group Meetings Log:

Students can record their meetings using the group meetings log.

Information:

Displays information about the people in a classroom.

2.2 Functional Requirements

From both instructors' and students' point of view:

Login to course: Entering the website link user is navigated to the login screen asking his/her email and password. User either logs in by entering his/her email and password or switches to sign up page. On sign up page users are required to enter his/her email, password and school.

Options: Users are able to modify their profiles: contact information, name etc.

Course page: Users have a courses page displaying the courses they signed up.

All groups page: Users have an all groups page displaying all groups in a course.

Display course & participant information: Users can display the course information and contact information of students and instructors.

Use private/group/general chat: Users can chat synchronously in three ways: person to person(private), as a group or as a course participant(general). Private chat is either between two students, two instructors or one student and one instructor. Group chat is between any students in a group. General chat is between all students and instructors in the course.

Use forums: Allows all users, students and instructors, to share questions and answers among each other.

Use calendar: Displays the deadlines for the assignments given by instructors.

Review Groups: All users are able to review each group in all groups page. Displays the participant list and information about the group.

View statistics for grades: All users are able to view the statistics computed by comparing the grades between groups(artifact-review) and instructors in Gradebook. Also displays assignment grades for groups, this is visible for participants in a particular group and instructors in the course.

Only from instructors' point of view:

Create a course: The course will be created by an instructor and a sign up link to the course will be provided to students.

Assignments page: Allows instructors to create assignments through assignments page. This assignment will be displayed on the calendar as an event. Instructors can upload a file and specify criterias.

View student dashboard: All instructors are able to see the detailed reviews about all students in the course.

View statistics for grades: In addition to viewing the statistics computed by comparing the grades between groups and instructors, also displays the detailed statistics regarding in-group peer review and assignment grades in Gradebook.

Use analytic tools for instructors: Allows to import-export grades. Import review questions to be used in in-group peer review and artifact review, in addition to the template questions provided by the tool.

Assign unassigned students to groups: Allows instructors to assign remaining students to groups that have available seats.

Only from students' point of view:

My Group Page: Students have my group page displaying group information, join requests, group chat, assignment upload link and calendar displaying deadlines for tasks created by group members and assignments given by instructors. Group members can create tasks to divide the workload among themselves and review the assignment file uploaded by the instructor.

Use calendar: In addition to displaying the deadlines for the assignments given by instructors, also displays the deadlines for the group tasks created through in-group work distribution tool.

Form a group: A student can create a group himself/herself and manage invitation, join and bundle me with friend requests. A student can send an invitation to others to join his/her group through an unassigned students list. He/she can accept or decline a join/bundle me with a friend request.

Find a group: All groups created are visible in all groups page. Students can review groups, if there are available seats and the student is not already in a group, he/she can send a join/bundle me with friend request to a group.

Use in-group work distribution tool: Group members are able to create their own tasks and specify deadlines on a calendar. They can specify the task and the responsible member.

Use group meetings notepad: Students can keep a record of the key points in a group meeting.

In group peer review: Enables group members to review and grade each other anonymously. This is only visible to person reviewing, not visible to group members or other groups.

Peer review of artifacts: Enables reviewing and grading other groups' work anonymously. This is only visible to group reviewing and grading, not visible to other groups.

Review and Upload assignment: Students can see the assignment file and upload the assignments given by the instructor through their group page.

2.3 Nonfunctional Requirements

Extendibility: Classroom Helper tool will have an extendible design to enable adding new features or extending the current features without having to modify the legacy code substantially. This efficiency will be achieved by employing Object-Oriented Programming principles.

Maintainability: Classroom Helper tool will have a maintainable design to repair or replace faulty or worn-out components without having to replace still working parts, maximize a product's useful life, make future maintenance easier and cope with a changing environment. It will be achieved by using plenty of comments and forming a clear and easy-to-understand structure for design and implementation.

Usability: Classroom Helper tool will have a simple and user-friendly interface supporting a range of user actions which will enable users to quickly grasp the use of the tool. User interface components and verbal interactions will be similar to that of contemporary communication tools. Names, messages, buttons, icons and graphics will require a basic and intuitive understanding so that they will be tailored to meet the naive users' requirements.

Reliability: The program should have a secure connection and should use HTTPS protocol. The program should have a little bit of security against data theft. Since none of the group members are experienced in web security, this cannot be a first priority for us. [Security Requirement]

The program should have a minimum 95% uptime (1 hour in a day might be down). [Mean time between failures requirement]

The system should backup once in 3 days to protect against power loss. [Data loss tolerance]

Performance Requirements: The System should support at least 50 concurrent users and ,to allow a smooth experience, all user inputs must be acknowledged within 5 second and the Classroom Helper tool must respond to a user action within 5 second.

2.4 Pseudo Requirements

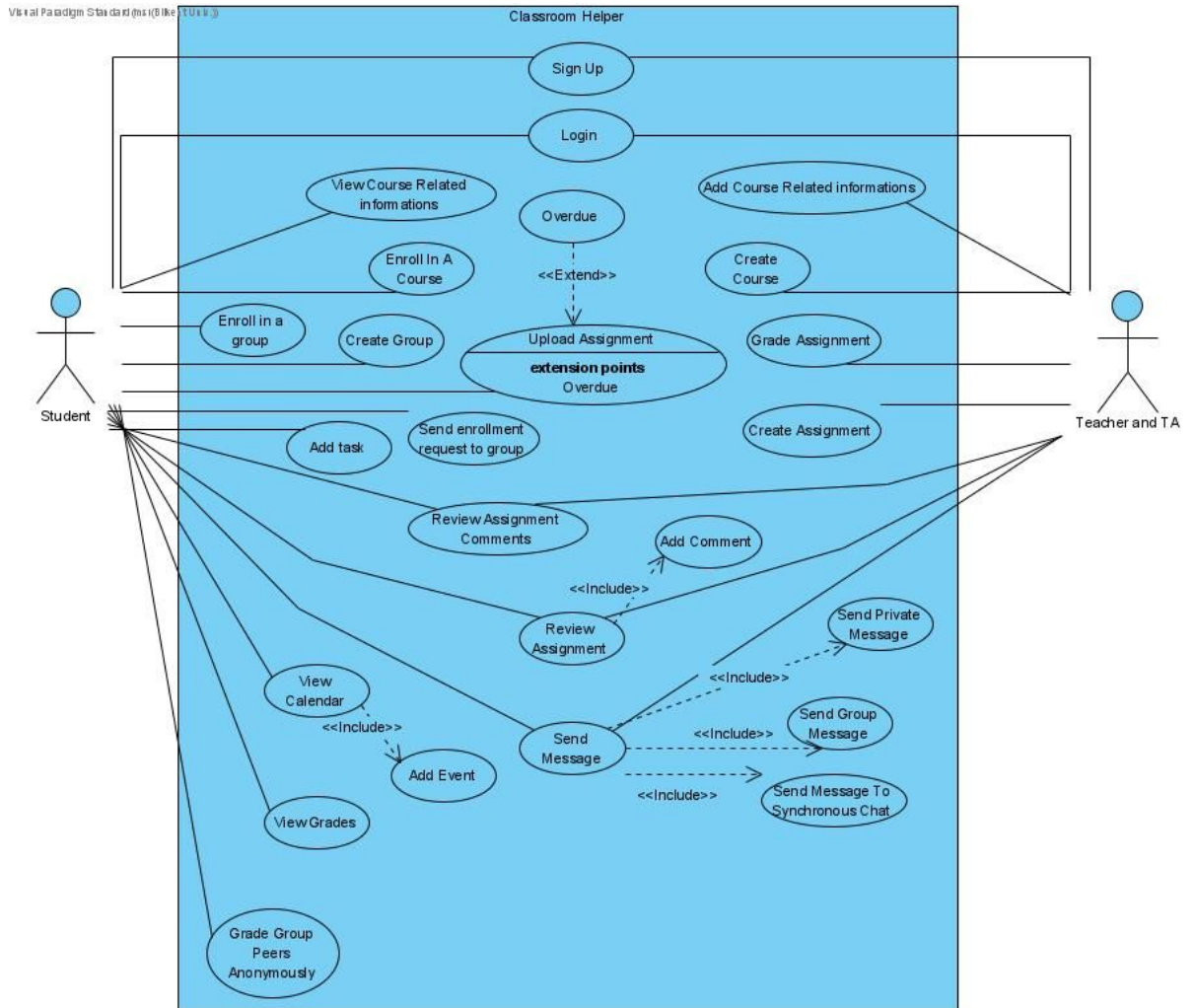
Implementation Requirements: Classroom Helper will use React for frontend, Java for backend and mySQL for database.

Client Software Requirements: Since the program will use React in frontend, the client's browser should support it. Chrome, Firefox, Safari should be the latest. Other platforms might work but will not be officially supported. Older versions of any kind of browser will not be supported because of time constraints.

Legal Requirements: The program must conform to KVKK (Kişisel verileri koruma kanunu - Data Protection Regulations).

3. System models

3.1 Use case model



Use Case #1

Use case name: Create Group.

Participating actors: Student.

Stakeholders and interests:

- Student press the create group button
- The system displays the create group screen

Pre-conditions:

- The student must be logged in to the system
- The student must not be in another group

Post-conditions:

- No postconditions

Entry-conditions:

- The student selects the create group button

Exit-conditions:

- Student selects the “cancel” button
- Student selects the “create the group” button

Success Scenario Event Flow:

1. Student selects the all groups button
2. System displays the all groups page
3. Student selects the create group button
4. System displays the create group page
5. Student fills out the necessary informations
6. Student clicks on the “create the group” button
7. System adds the new group to the all groups page

Alternative Event Flows:

- 1) If student wants to cancel the create group process
 - a) Student selects all groups
 - b) System displays all groups
 - c) Student selects the create group button
 - d) System displays the create group page
 - e) Student clicks on the “cancel”
 - f) System displays the all groups page

Use Case #2

Use case name: Send enrollment request to a group.

Participating actors: Student.

Stakeholders and interests:

- Student press the send enrollment request button
- System adds the request to group page

Pre-conditions:

- The student must be logged into the system
- Student must not be in another group

Post-conditions:

- Student added to the group as a member

Entry-conditions:

- Student press the send enrollment request button

Exit-conditions:

- No exit conditions

Success Scenario Event Flow:

1. Student selects the all groups button
2. System displays the all groups page
3. Student selects the send enrollment request button

4. System adds the request to the group page

Alternative Event Flows:

1. No alternative event flow

Use Case #3

Use case name: Upload Assignment.

Participating actors: Student.

Stakeholders and interests:

- Student press the upload assignment button
- System shows the uploaded file on the corresponding group page

Pre-conditions:

- The student must be enrolled in a group
- The assignment must be uploaded before the due date.

Post-conditions:

- No postconditions

Entry-conditions:

- Student selects upload assignment button

Exit-conditions:

- Student selects the “cancel” button
- Student selects the “upload the file” button

Success Scenario Event Flow:

1. Student selects the all groups button
2. System displays the all groups page
3. Student clicks on him/her group page
4. System displays the group page

5. Student selects upload assignment button that next to the corresponding assignment
6. System displays the upload assignment page
7. Student selects the file
8. Student presses the “upload the file” button
9. System displays the file on the group page

Alternative Event Flows:

- 1) If student wants to cancel the upload
 - a) The student selects my groups
 - b) System displays the group page
 - c) Student selects upload assignment button that next to the corresponding
 - d) System displays the upload assignment
 - e) Student selects the file
 - f) Student presses the “upload the file”
 - g) System displays the file on the group page
- 2) If the student wants to cancel the upload process
 - i) Student selects the my group button
 - ii) System displays the group page
 - iii) Student selects upload assignment button that next to the corresponding assignment
 - iv) System displays the upload assignment page
 - v) Student clicks the cancel button
 - vi) System displays the group page

Use Case #4

Use case name: Review Assignment.

Participating actors: Student, Teacher and TA.

Stakeholders and interests:

- Student, or Teacher or TA press the file icon
- System displays the uploaded assignment

Pre-conditions:

- Student, or Teacher or TA must be on the corresponding group page
- There must be an uploaded file for an assignment.

Post-conditions:

- No postconditions

Entry-conditions:

- Student, or Teacher or TA press the file icon

Exit-conditions:

- Student, or Teacher or TA should press the exit button

Quality requirements:

At any point during the flow of events, this use case includes the Add comment use case. This Add comment use case is initiated when the Student, Teacher or TA invokes the add comment function that will be also displayed on the review assignment page. When this use case is invoked, the system will add the comment on the file.

Success Scenario Event Flow:

1. Student, Teacher or TA selects the all groups button
2. System displays the all groups page
3. Student, or Teacher or TA clicks on the file icon
4. System displays the uploaded assignment
5. Student, or Teacher or TA presses the exit button
6. System returns to the group page.

Alternative Event Flows:

1. No alternative event flow.

Use Case #5

Use case name: Grade group peers anonymously.

Participating actors: Students.

Stakeholders and interests:

- Student press the grade your peer button
- The system displays the peer grade screen

Pre-conditions:

- The student must be in a group

Post-conditions:

- No postconditions

Entry-conditions:

- Student clicks the grade your peer button

Exit-conditions:

- Student selects the “cancel” button
- Student selects the “submit” button

Success Scenario Event Flow:

1. Student selects the all groups button
2. System displays the all groups page
3. Student clicks on his/her group page
4. System displays the group page
5. Student clicks the grade your peer button
6. System displays the grade your peer page
7. Student gives the grades of his/her peers
8. Student clicks on the “submit” button
9. System displays my group page

Alternative Event Flows:

1. If student wants to cancel the peer grading process
 - a. Student selects the all groups button System displays the all groups page
 - b. Student clicks on his/her group page
 - c. System displays the group page

- d. Student clicks the grade your peer button
- e. System displays the grade your peer page
- f. Student clicks on the cancel button.
- g. System displays my group page

Use case name: Grade Assignment.

Participating actors: Teacher and TA.

Stakeholders and interests:

- Teacher and TAs press the grade assignment icon
- The system displays the grading screen

Pre-conditions:

- Teacher and TA must be on a group page
- There must be an uploaded assignment

Post-conditions:

- No postconditions

Entry-conditions:

- Teacher or TA press the grade assignment icon

Exit-conditions:

- Teacher or TA press the “cancel” button
- Teacher or TA press the “upload the grade” button

Success Scenario Event Flow:

1. Teacher or TA press the all groups button
2. System displays the all groups page
3. Teacher or TA went to a group page by pressing on it
4. System displays the group page
5. Teacher or TA press the grade assignment button
6. System displays the grading page

7. Teacher or TA gives enters the grade
8. Teacher or TA press the upload the grade button
9. System displays the group page

Alternative Event Flows:

- 1) If a Teacher or TA wants to cancel the create a group process
 - a) Teacher or TA press the all groups button
 - b) System displays the all groups page
 - c) Teacher or TA went to a group page by pressing on it
 - d) System displays the group page
 - e) Teacher or TA press the grade assignment button
 - f) System displays the grading page
 - g) Teacher or TA presses the cancel button
 - h) System displays the group page

Use Case #7

Use case name: Send Private Message.

Participating actors: Student, Teacher and TA.

Stakeholders and interests:

- Student, Teacher and TAs wrote a message and press the send button
- The system displays the message in the chat box

Pre-conditions:

- Student, Teacher or TA must be enrolled in a course

Post-conditions:

- No postconditions

Entry-conditions:

- Student, Teacher or TA press chat icon that is next to the users' name

Exit-conditions:

- Student, Teacher or TA press the send button

Success Scenario Event Flow:

1. Student, Teacher or TA press the people button
2. System displays the people page
3. Student, Teacher or TA press the chat icon that is next to another user's name
4. System displays private chatbox
5. Student, Teacher or TA wrote a message and press the send button
6. System shows the message in the private chat box

Alternative Event Flows:

1. If Student, Teacher or TA wants wrote a message in the group page
 - a. Student, Teacher or TA press all groups button
 - b. System displays the all group page
 - c. Student, Teacher or TA presses the group
 - d. System displays the group page
 - e. Student, Teacher or TA press the chat icon that is next to the another user's name
 - f. System displays private chatbox
 - g. Student, Teacher or TA wrote a message and press the send button
 - h. System show the message in the private chatbox

Use Case #8

Use case name: Send Group Message.

Participating actors: Student

Stakeholders and interests:

- Student wrote a message and press the send button
- The system displays the message in the group chat

Pre-conditions:

- Student must be enrolled in a group

Post-conditions:

- No postconditions

Entry-conditions:

- Student, goes into the my group page

Exit-conditions:

- Student press the send button

Success Scenario Event Flow:

1. Student, press the my group button
2. System displays the group page
3. Student wrote a message to the group chat and press the send button
4. System shows the message in the group chat

Alternative Event Flows:

1. If Student, Teacher or TA wants to write a message on the group page
 - a. Student press the all groups button
 - b. System displays the all group page
 - c. Student presses his/her group
 - d. System displays the group page
 - e. Student wrote a message to the group chat and press the send button
 - f. System shows the message in the group chat

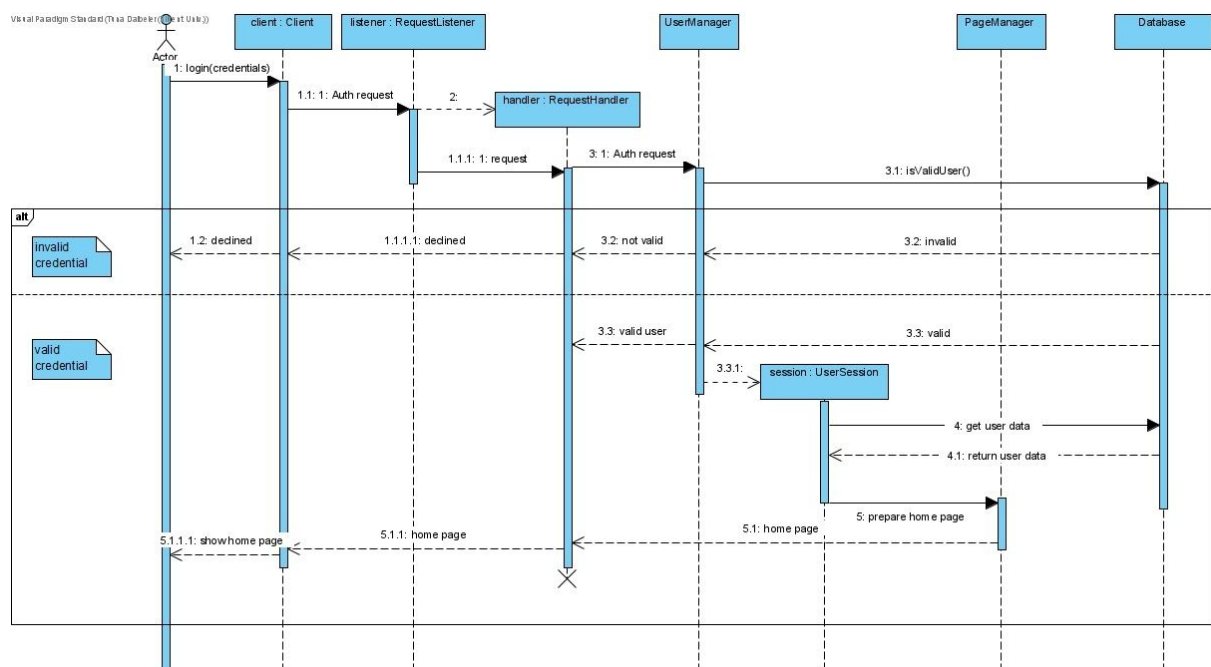
3.2 Sequence Diagrams

Please note that some of the classes shown in the sequence diagrams are not final versions and they are not added to the class diagrams because some of them are implementation details. The implementation details are not included in the analysis report. The frameworks and languages are not finalised even though we have decided to use React and Java and our knowledge about these are not comprehensive. Because of this, a very shallow explanation is written. For more information please follow the link:

[Client-Server Architectures](#)

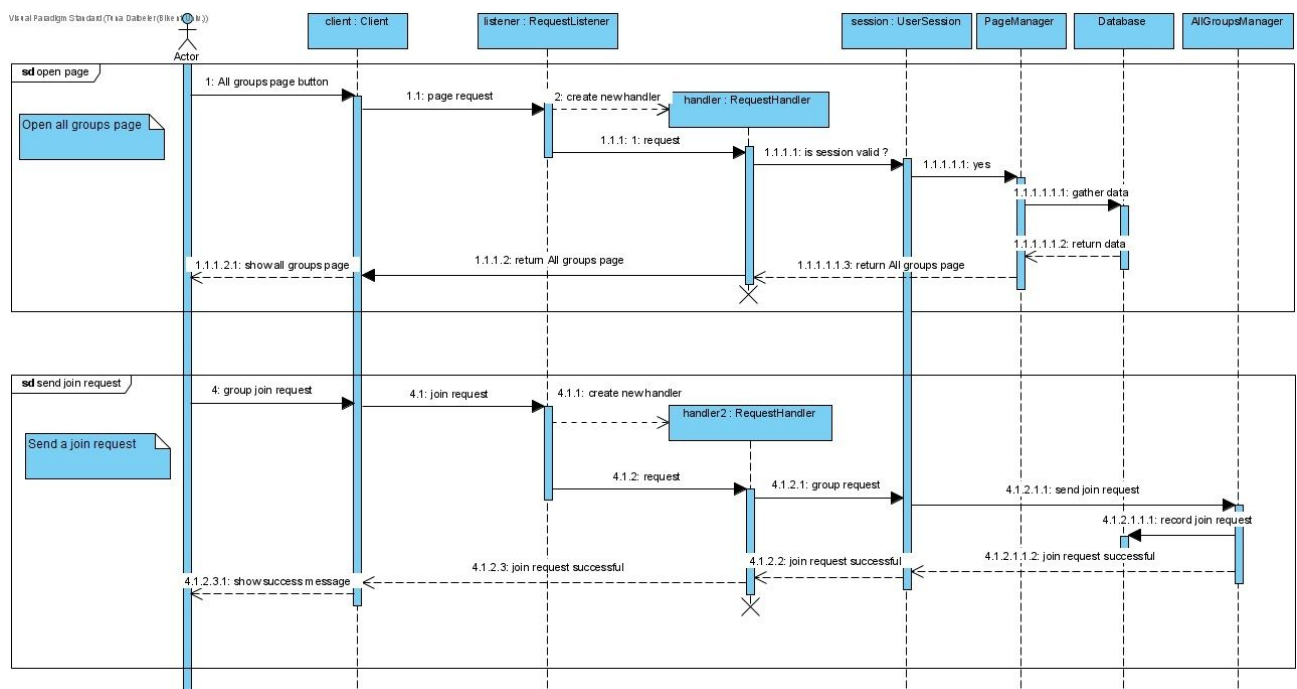
Scenario 1: Login

The user enters their credentials to the client. The Client creates a request and sends the email and password to the backend. RequestListener processes the request and creates a request handler. This RequestHandler creation process' purpose is to handle more than one request in parallel. Handler sends credentials to UserManager and UserManager checks if the user exists in the database. If user email - password does exist in the database sends the user a declined message. If it exists create a UserSession. UserSession holds the data about what the user did and auth token etc. Then PageManager prepares the homepage with users data. Then the homepage returns to the client.



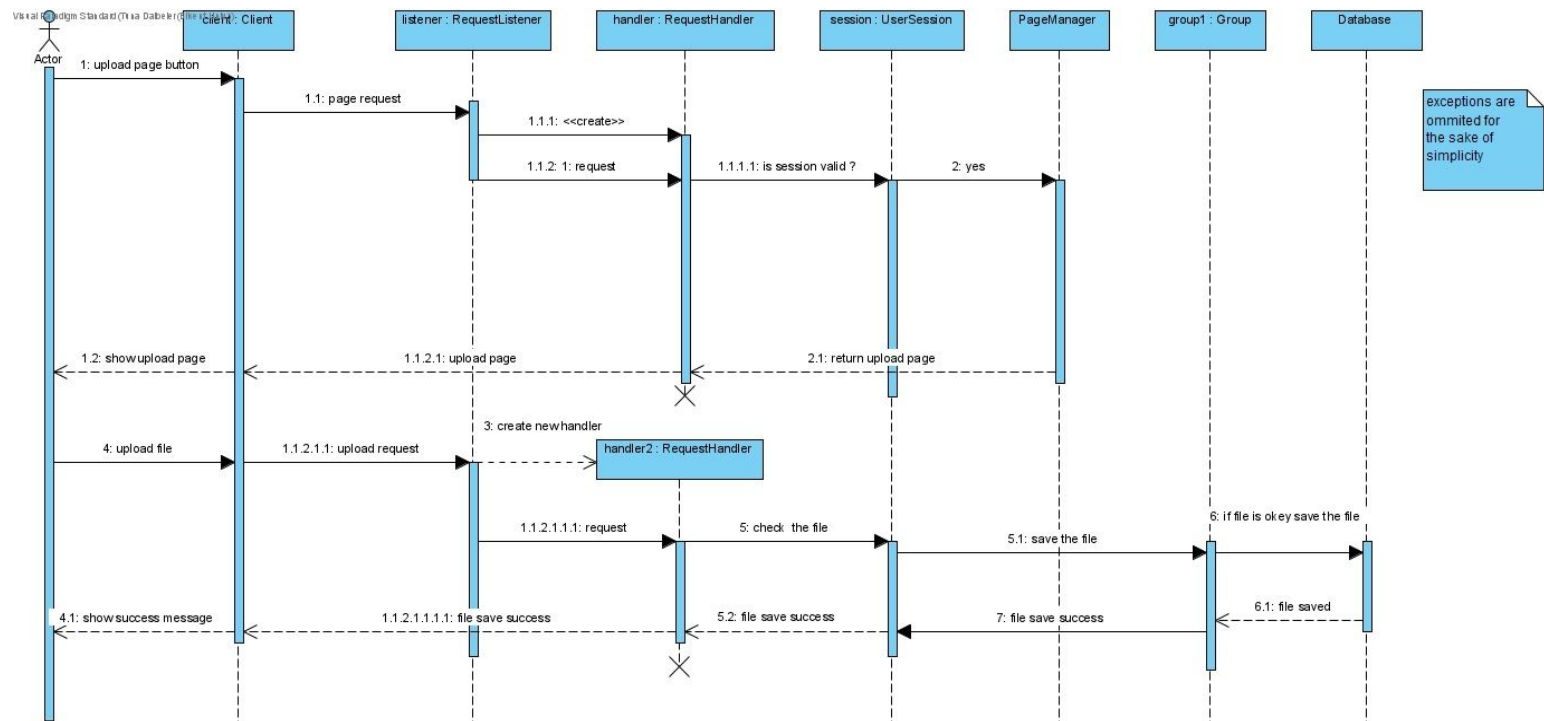
Scenario 2: Join a Group

The user clicks the “All Groups” button. A page request sent to the server. UserSession checks if the user is eligible to access the page. If not, return an error. Error process is omitted from the diagram for the sake of simplicity. If the request is okay send the page to the user. On the All Groups page, the user clicks join and the join request is sent to the server. The handler passes the request to UserSession. UserSession makes necessary checks (like if the user is already in a group or not). Then UserSession passes the request to AllGroupsManager. AllGroupsManager is the class that holds all groups and records groups history, also handles adding and deleting a group, also checks if the group choosing period is over or not (Group choosing period is when people can freely form groups. Later will be disabled by the instructor or an admin account.). If the request is okay, records the request into the database and returns a successful message.



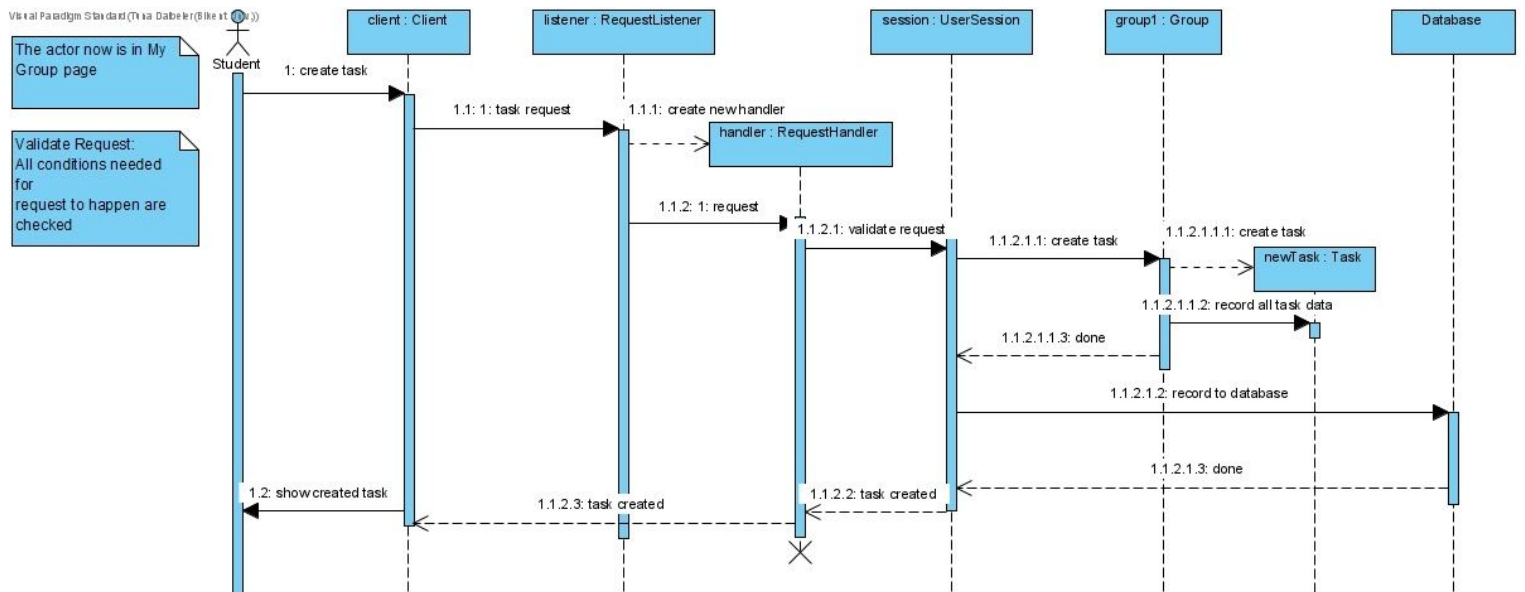
Scenario 3: Upload Assignment

After the user got into the upload page. User clicks upload file and chooses a file and sends an upload request. After that file, passed through Handler and UserSession, reaches the user's group. The Group records the file into the system.



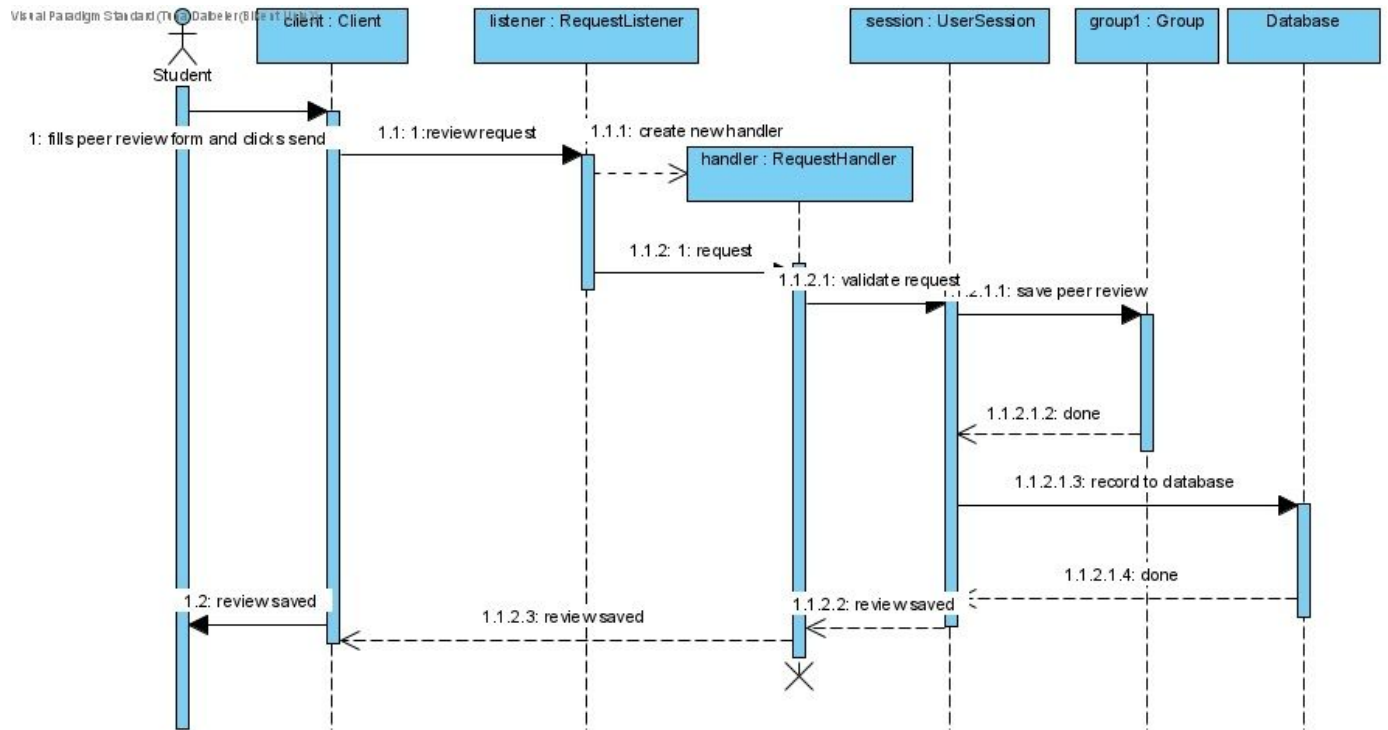
Scenario 4: Tasks

Assumed the user is on the group page now. The student clicks the create task button and fills the form opened on the page. In the form, the student enters task info (who is assigned to the task, what is the due date, name of the task and details etc.) Then, the task request is sent to the server. The server passes the task info to a group class. Group class opens a new task object and records info to the object. Then the task gets recorded to the database and returns the user to a success message.



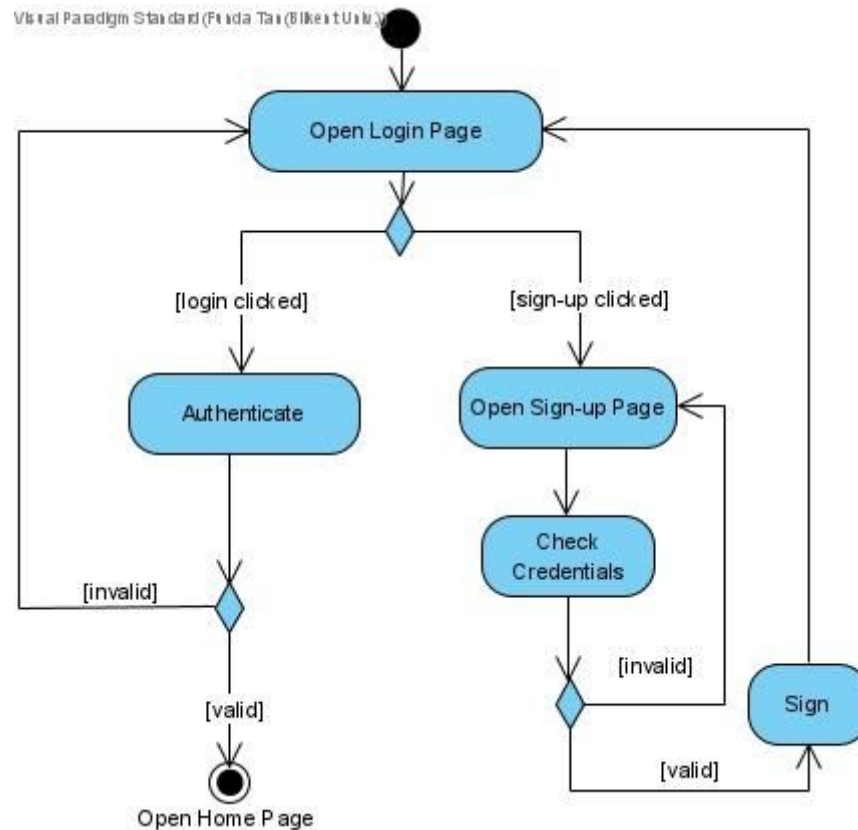
Scenario 5: In Group Peer Review

Same process with a couple of scenarios. A form sent from the client recorded into the database.



3.3 Activity Diagrams

Activity Diagram 1: Login



When the user opens the system, the login page is displayed. Users can be students or teachers at this point.

Case 1.1: If the login is clicked, the system will authenticate the user.

Case 1.1.1: If the authentication is invalid, system displays login page again.

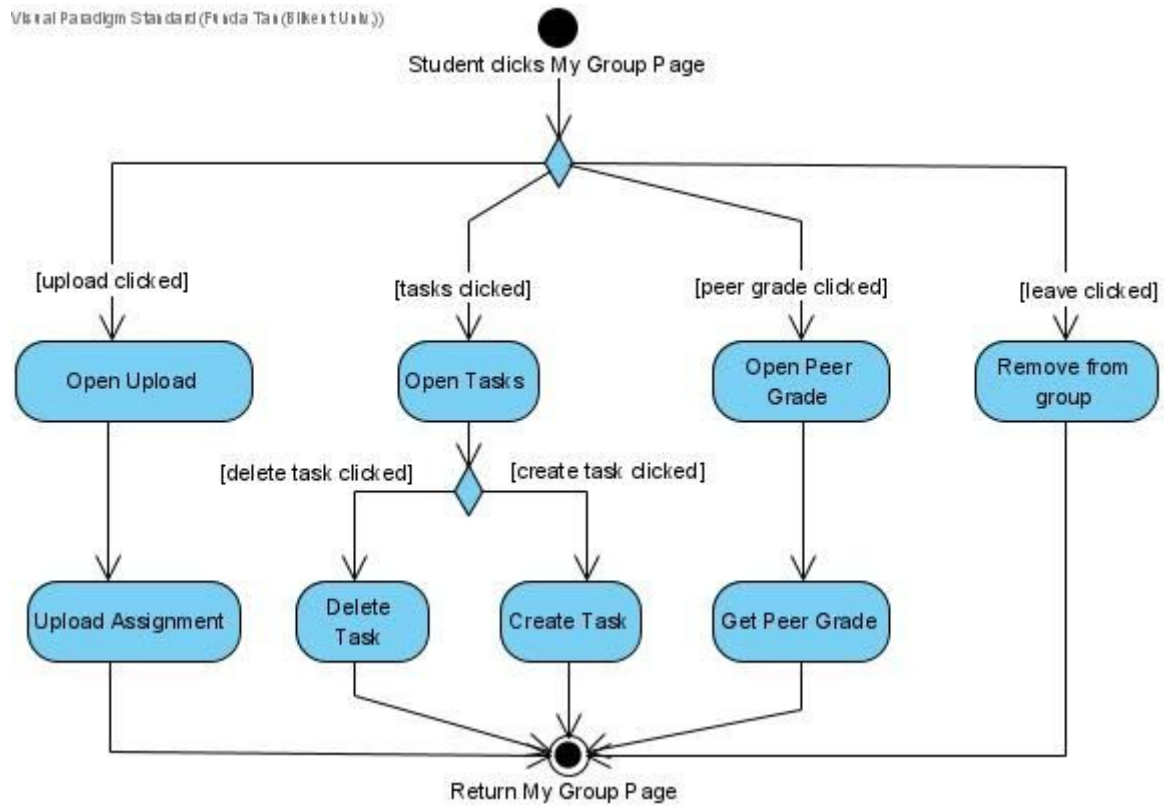
Case 1.1.2: If the authentication is valid, system open "Home Page".

Case 1.2: If the sign-up is clicked, the system will display the sign-up page then checks the credentials.

Case 1.2.1: If credentials are invalid, system displays the sign-up page again.

Case 1.2.2: If the credentials are valid, the system opens the login page again.

Activity Diagram 2: My Group Page



When the student clicks “My Group” page, there are four cases possible.

Case 1: If upload is clicked, the upload page will be opened. Then, the system uploads the assignment and returns the “My Group” page.

Case 2: If “tasks” is clicked, the tasks page will be opened.

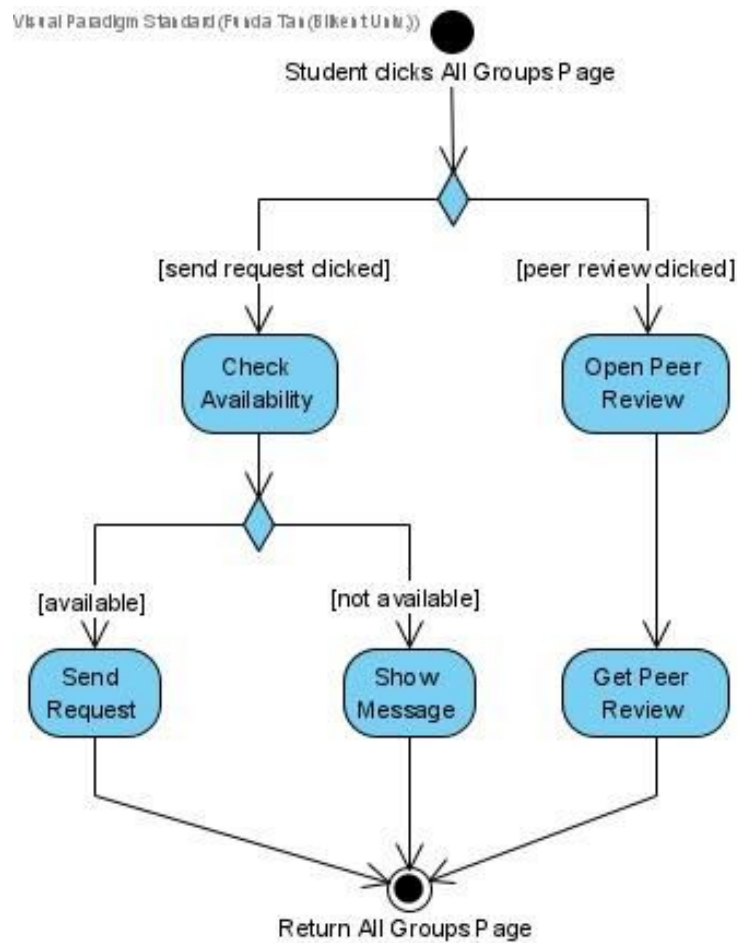
Case 2.1: If “delete task” is clicked, the system will delete the task and return the “My Group” page.

Case 2.2: If “create task” is clicked, the system will delete the task and return the “My Group” page.

Case 3: If peer grade is clicked, the peer grade page will be opened. Then, the system gets the peer grade and returns the “My Group” page.

Case 4: If leave is clicked, the system removes the student from the group and returns the “My Group” page.

Activity Diagram 3: All Groups Page (Students)



When the student clicks “All Groups” page, there are two cases possible.

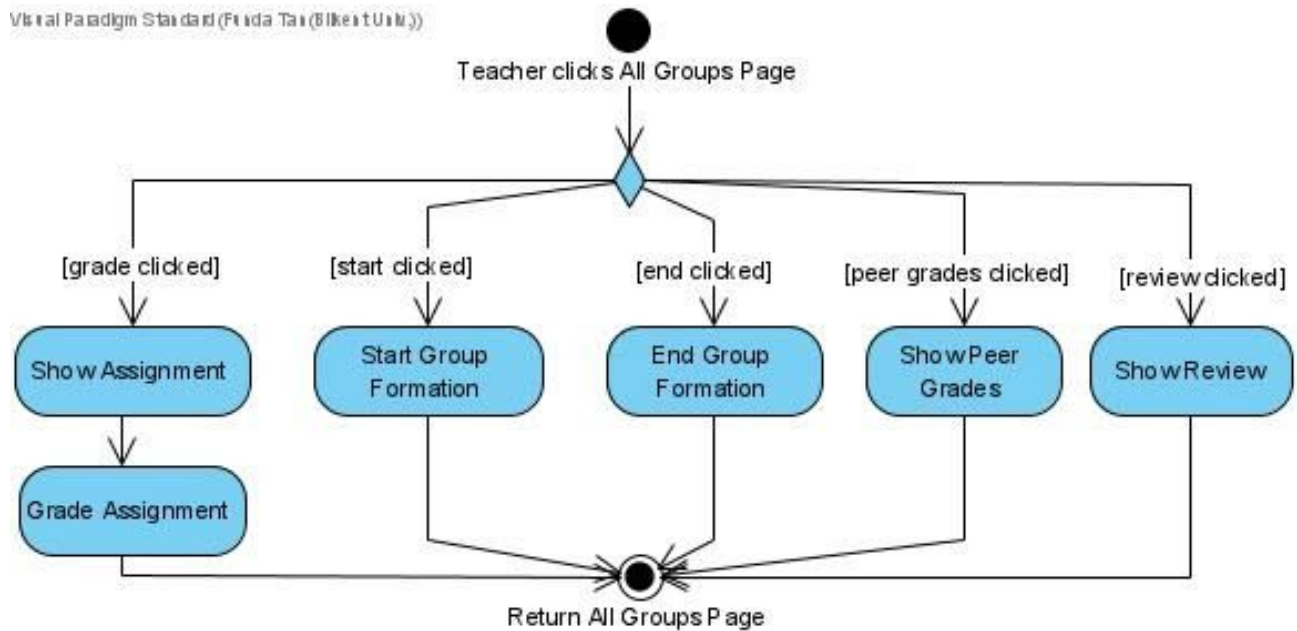
Case 1: If “send request” is clicked, system checks availability.

Case 1.1: If the group is available, a request is sent from the user to the group and returns to the “All Groups” page.

Case 1.2: If the group is full, system shows a message indicating the unavailability and returns to “All Groups” page

Case 2: If peer review is clicked, system opens peer review, then gets peer review and returns to “All Groups” page

Activity Diagram 4: All Groups Page (Teachers)



When the teacher clicks “All Groups” page, there are five cases possible.

Case 1: If grade is clicked, system shows assignment to grade and gets the grade and returns to “All Groups” page

Case 2: If start is clicked, system starts the group formation process and returns to “All Groups” page.

Case 3: If the end is clicked, the system ends the group formation process and returns to the “All Groups” page.

Case 4: If peer grades is clicked, system shows peer grades and returns to “All Groups” page.

Case 5: If review is clicked, system shows the review and returns to “All Groups” page.

Visual Paradigm Standard (Ezgi Saygılı (Bilkent Univ.))



Answer class: A subclass of Post class. Defines an answer to be used in Forum.

Question class: A subclass of Post class. Defines a question to be used in Forum.

Post class: Defines a post with text, time, contributor, status and post number to be used in Forum. Specifies time as Date and contributor as People.

Forum class: Defines a forum platform which enables sharing Questions and Answers as Posts.

Instructor_Constraints class: Defines constraints/limits for Instructor users.

Student_Constraints class: Defines constraints/limits for Student users.

Instructor class: A subclass of People class. Defines user as Instructor.

Student class: A subclass of People class with an addition of Grade specification. Defines user as Student.

People class: Users are people. Defines People with name, contact information and ID.

Group class: People form groups. Defines a Group with name, group number and a set of People.

Course class: Course consists of groups. Defines a Course with name, course number and a set of Groups.

Date class: Defines a date with day, month and year to be used in events.

Event class: Defines an event with name and time. Specifies time as Date.

Task class: Subclass of Event class with an addition of number and status specifications.

Calendar class: Defines a calendar which enables to modify and display Events.

TemplateQuestion class: Defines a template question with a question and a set of choices to be used in peer review and artifact review.

PeerGrade class: A subclass of Grade class. Defines the grade of a student given through peer review to be used in Gradebook.

ArtifactGrade class: A subclass of Grade class. Defines the grade of a group given through artifact review to be used in Gradebook.

Grade class: Defines a grade with a score and comments.

Gradebook class: Defines a student dashboard platform. Defines the following operations:.

1) Provides artifact review functionality. Enables groups to review and grade each other anonymously.

2) Provides in-group peer review functionality. Enables group members to review and grade each other anonymously.

3) Provides statistics by comparing Grades given through artifact review with imported grades given by Instructor.

4) Contains template review questions and imports new review questions from the user file provided.

Group_Request class: A subclass of Request class with an addition of receiver as Group specifications.

In_Person_Request class: A subclass of Request class with an addition of receiver as People specifications.

Request class: Defines a request with a text, request number, status and sender to be used in grouping. Specifies sender as People.

File_Manager class: Saves and load data from a file.

Input_Manager class: Gets input from the user.

Menu class: Menu is fixed at the leftmost part of the screen. Defines the following operations:

- 1) Going back to the course page
- 2) Switching to chat platform
- 3) Switching to my group page
- 4) Switching to all groups page
- 5) Switching to Gradebook
- 6) Display People
- 7) Display Options
- 8) Switching to Forum page

People_Manager class: Manages People by adding, removing, modifying and displaying the People.

Group_Manager class: Manages a Group by adding, removing, modifying and displaying the Group and group Tasks. Also it enables general Tasks/assignments to be uploaded.

All_Groups_Manager class: Manages Groups by defining the following operations:

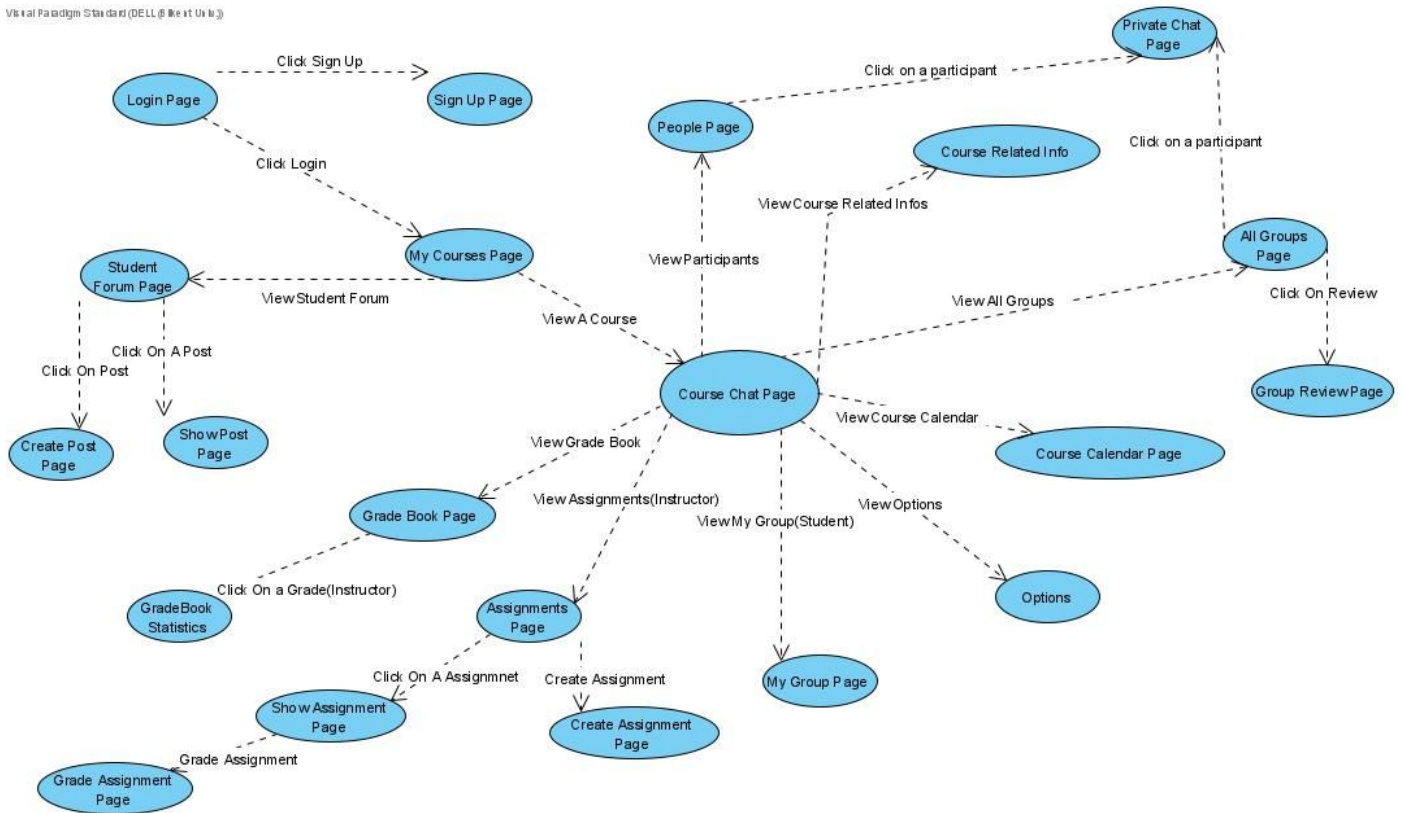
- 1) Display all groups
- 2) Review a Group
- 3) Send join Request to a Group
- 4) Send invitation Request to People
- 5) Send bundle me with friend Request to a Group
- 6) Accept/decline a Request
- 7) Leave a Group
- 8) Display unassigned Students
- 9) Assign remaining Students

Course_Manager class: Manages a Course by adding, removing, modifying and displaying the Course and general Tasks.

3.5 User interface - navigational paths and screen mock-ups

3.5.1 Navigational Paths

Visual Paradigm Standard (DELL & Weixi University)



3.5.2 Screen Mock-Ups

3.5.2.1 Login Page

Login

Your Email:

Password:

Sign Up

Login

The initial page is the login page. If the sign up button is clicked, then the sign up page opens, if the user enters valid email and password, then the home page opens up.

3.5.2.2 Sign Up Page

Sign Up

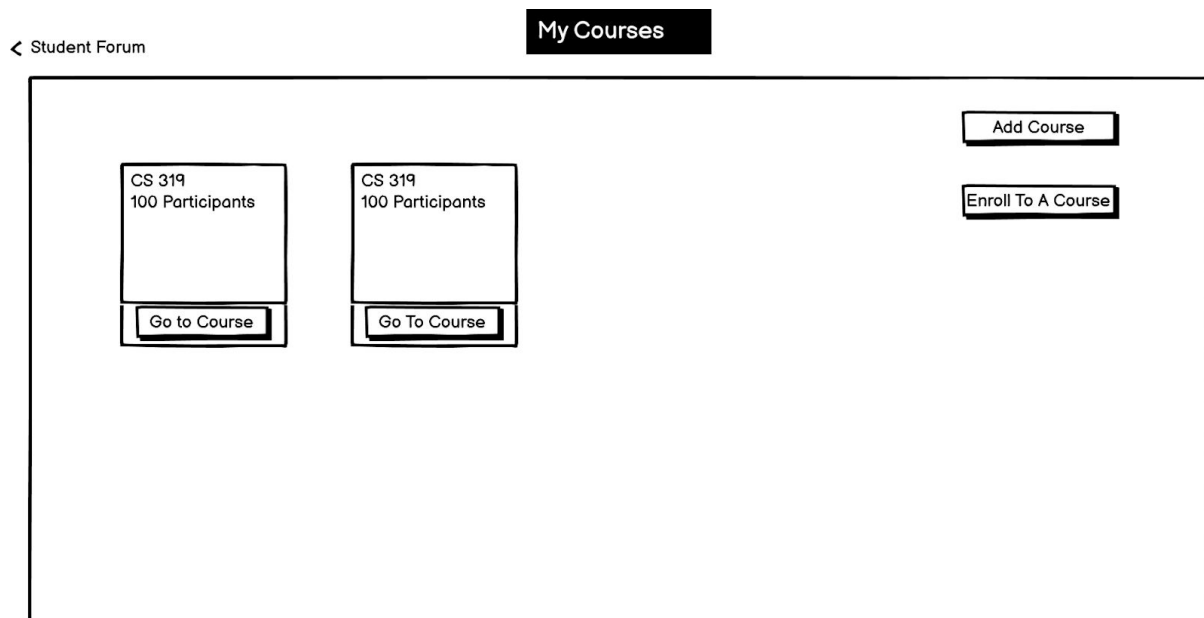
Your Email:

Password:

Your School:

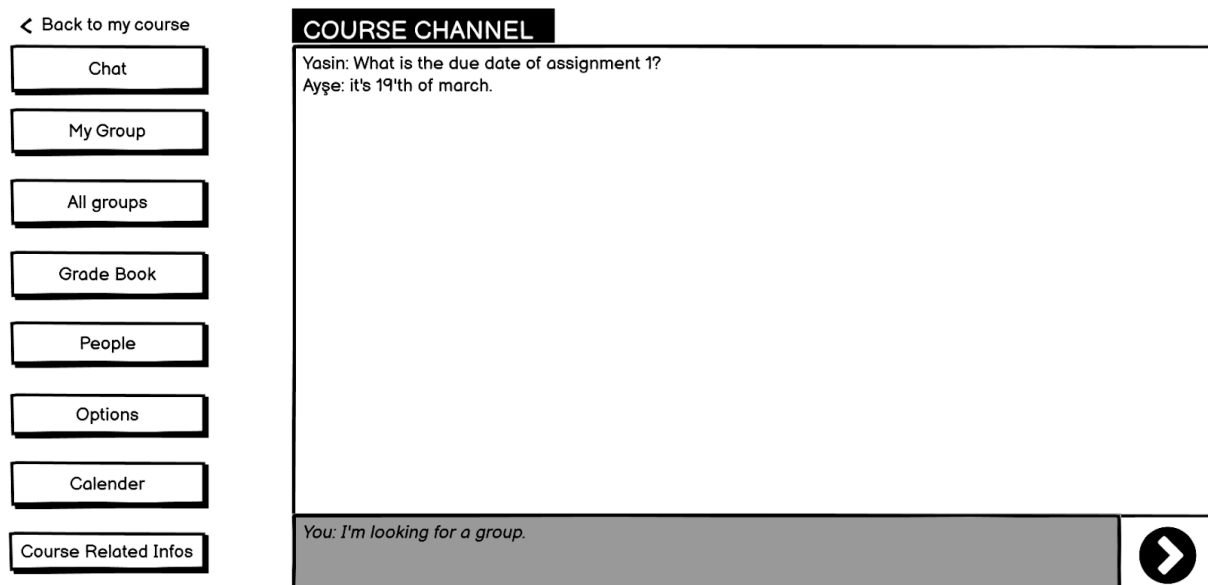
Sign Up

3.5.2.3 My Courses Page



This is the homepage for the application. It shows enrolled courses. All courses have buttons which direct the user to the course chat page. There are two buttons which enable creating a course or enrolling to a specific course. Users can switch to the student forum by student forum button.

3.5.2.4 Course Chat Page



This is the page where the users interact with the synchronous chat of course. There is a menu for all course related pages which enables transitions between pages. This menu has eight buttons Chat, My Group(Assignments for instructors), All Groups, Grade Book, People, Options, Calendar and Course Related info. Also, there is a button which allows the user to return to the home page. The Course Chat page is where the Chat button is linked.

3.5.2.5 My Group Page

[Back to my course](#)

Chat

My Group

All groups

Grade Book

People

Options

Calendar

Course Related Infos

MY GROUP

Group 1

GROUP MEMBERS

Ali
Ayşe
Veli
Fatma
Yasin

MARCH 2021

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

Add Event To Group Calendar

Grade your peer

JOIN REQUESTS

Sıla wants to join
Mehmet wants to join
Ahmet wants to join
Beril wants to join

☒ ☒ ☒
☒ ☒ ☒
☒ ☒ ☒
☒ ☒ ☒

Assignment 1

Assignment 2

Yasin: Ayşe can you create the use case diagram?

Ayşe: yes.

You: Let's meet on wednesday at 9 pm via zoom.

My Group page is where the students interact with his/her group. The student can see group mates, tasks, task calendar, assignments, join requests and group chat on this page. Students upload given assignments and organize group tasks on this page.

3.5.2.6 All Groups Page

[Back to my course](#)

Chat

My Group

All groups

Grade Book

People

Options

Calendar

Course Related Infos

ALL GROUPS

Group 1

Ali
Ayşe
Veli

2 remaining

Send join request

Review

Group 2

Fatma
Yasin

3 remaining

Send join request

Review

Add group

Unassigned Students

Mehmet
Sıla

This is the page where users can interact with groups of the course. They can see created groups, their members and unassigned students. Students can create groups by the Add group button if they did not join a group. Also, every group has two buttons: Send join request and Review. Send join button creates a join request to join that group. Review opens a page which shows the work of the group which will be reviewed.

3.5.2.7 Group Review Page

The screenshot shows the 'Group Review Page' interface. On the left is a sidebar with a 'Back to my course' link and buttons for 'Chat', 'My Group', 'All groups', 'Grade Book', 'People', 'Options', 'Calender', and 'Course Related Infos'. The main content area has a header 'ALL GROUPS' and a section for 'Group 1'. Inside this section, there are two review items: 'To be Group Reviewed 1' and 'Group Review 1', each with a speech bubble icon. An 'Upload Review' button is located at the bottom right of the main content area.

This page allows students to get other groups' to be reviewed assignments and review them. Uploading a review is done by the Upload Review button.

3.5.2.8 Grade Book page

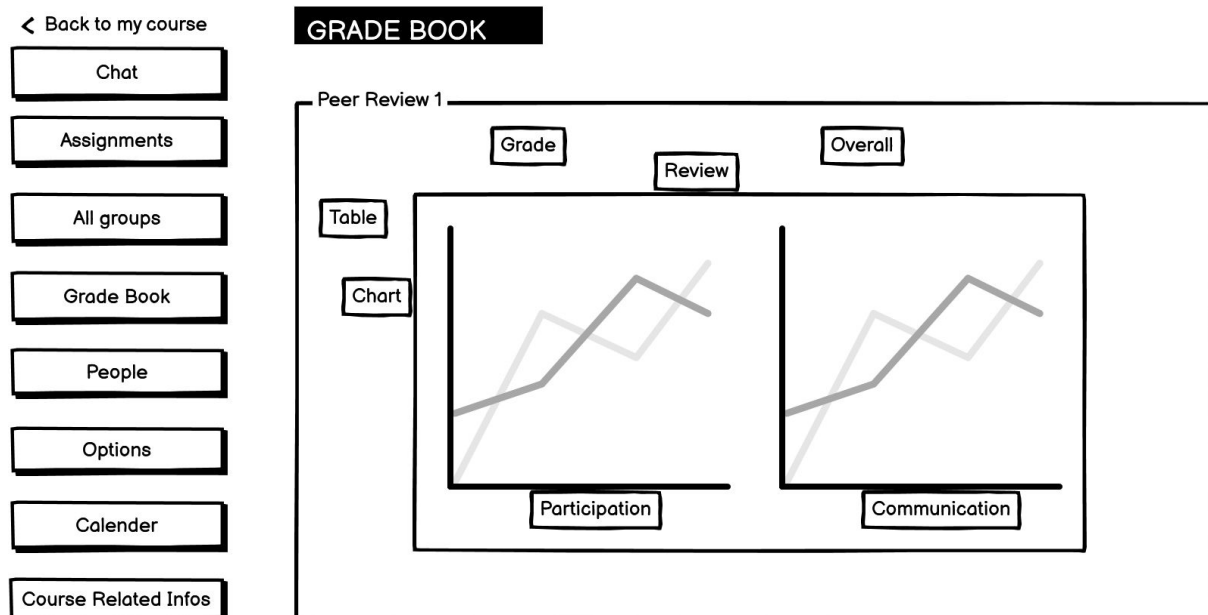
The screenshot shows the 'Grade Book' page interface. On the left is a sidebar with a 'Back to my course' link and buttons for 'Chat', 'My Group', 'All groups', 'Grade Book', 'People', 'Options', 'Calender', and 'Course Related Infos'. The main content area has a header 'GRADE BOOK' and a table of graded assignments.

Assignment 1	98/100
Assignment 2	88/100
Peer review grade	90/100

This page shows graded assignments. Students will see grades they get from their assignments. Instructors will see a statistics button for every assignment and when clicked application will direct the instructor to the statistics page of the assignment.

3.5.2.9

Grade Book page continues (Instructor view)



Statistics page shows statistics of grades of students. There may be several options for demonstration of data.

3.5.2.10 People Page

[Back to my course](#)
[Chat](#)
[My Group](#)
[All groups](#)
[Grade Book](#)
[People](#)
[Options](#)
[Calender](#)
[Course Related Infos](#)

PEOPLE

Instructors

Eray Tüzün
Uğur Doğrusöz

Teaching Assistants

Elgun Jabrailzade

Students

A B
C D
E F

This page shows participants of the course in different categories such as instructors and TAs. Users can click on the names and message them. When clicked, they will be directed to the Private Chat Page.

3.5.2.11 Private Chat Page

< Back to my course

Chat
My Group
All groups
Grade Book
People
Options
Calender
Course Related Infos

Eray Tüzün

Hocam, should we upload requirements report to Github?

Yes

Thank you

This page allows the user to message with other participants of the course.

3.5.2.12 Course Calendar Page

< Back to my course

Chat
My Group
All groups
Grade Book
People
Options
Calender
Course Related Infos

CALENDER

MARCH 2021

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

This page shows event dates for the course as assignment due dates, task due dates.

3.5.2.13 Course Related Info

[< Back to my course](#)
Chat
My Group
All groups
Grade Book
People
Options
Calender
Course Related Infos

COURSE RELATED INFORMATION
[Course website](#)
Instructor: Eray Tüzün
Office Hours: EA 501 by appointment
TAs:
1) Elgun Jabrayilzade: elgun@bilkent.edu.tr, GitHub, EA 527 by appointment. Git Tutorial Lab, Slack, Visual Paradigm Accounts, Design report questions
2) Erdem Tuna: erdem.tuna@bilkent.edu.tr, GitHub, EA 527 by appointment. Requirement Report, Implementation and Final Report questions
3) Barış Ardiç: baris.ardic@bilkent.edu.tr, GitHub, EA 527 by appointment. Code Review lab
Grading:
Attendance/Quiz/Lab Assignments 20
Project 40
Midterm closed book & notes 15
Final closed book & notes 25

This page shows the information about the course and contact information of instructors.

3.5.2.14 Assignments Page (Instructor view)

[< Back to my course](#)
Chat
Assignments
All groups
Grade Book
People
Options
Calender
Course Related Infos

Assignments
Assignment 1 [See uploaded assignments](#)
Assignment 2 [See uploaded assignments](#)
Peer review grade [See uploaded assignments](#)
[Create assignment](#)

For instructors of course, there is the Assignments section instead of My Groups section in the menu. The instructor can create and reach assignments. When the Create Assignment button is clicked, the Create Assignment Page opens. Also, every assignment has their own "See Uploaded Assignments" which is linked to "Show Assignment Page".

3.5.2.15 Create Assignment Page

Back to my course

Chat

Assignments

All groups

Grade Book

People

Options

Calendar

Course Related Infos

Assignments

Create Assignment

Name of Assignment :

Assignment 1

Explanation For Assignment:

You Need to ...

Due Date :

/ /

Type of Assignment :

Peer Review

File:

Upload

Grading Criterias

Readability 40 / 40

Add Criteria

Review Criterias

Evaluate partner 5 Stars

Add Criteria

Finish

This page allows the instructor to create an assignment. The instructor may specify important characteristics of the assignment such as name, due date, explanation for the assignment and also if there is a pdf file for the assignment. Also, the instructor may specify the type of the assignment, such as individual homework, peer review or group review. Then there could be criterias which will be added by the instructor for easy grading and demonstrating the grades. The types of criterias would be grading criterias which will allow the instructor to grade the assignment in an organized way. Also, there are review criterias for review assignments which will allow students to review each other in an organized way. After clicking finish, assignment will be created and the application will return to "Assignments Page".

3.5.2.16 Show Assignment Page

Back to my course

Chat

Assignments

All groups

Grade Book

People

Options

Calendar

Course Related Infos

Assignments

Assignment 1

Selahattin Cem Öztürk

Grade

Funda Tan

Grade

Ezgi Saygılı

Grade

This page shows grades of students and if they are not graded, there will be a Grade button which will direct the instructor to “Grade Assignment Page”. Also, the instructor will be able to get uploaded files for the assignment by clicking on names.

3.5.2.17 Grade Assignment Page

[Back to my course](#)

Chat

Assignments

All groups

Grade Book

People

Options

Calendar

Course Related Infos

Assignment 1

Onur Korkmaz

Grading Criterias

Readability

39 / 40

Add Feedback

This page allows the instructor to grade assignments. The page will show criterias that the instructor attached to the assignment as grading criterias. The instructor will be able to specify grades. Also, the instructor may add feedback for the grade.

3.5.2.18 Student Forum Page

[Back to my course](#)

Student Forum

Q Search Something

Add Tag

Post

12 Answers

What is object diagram in UML?

CS 319 UML

1 Answers

What is $\int \frac{1}{1+x^2} dx$?

3 Answers

...

4 Answers

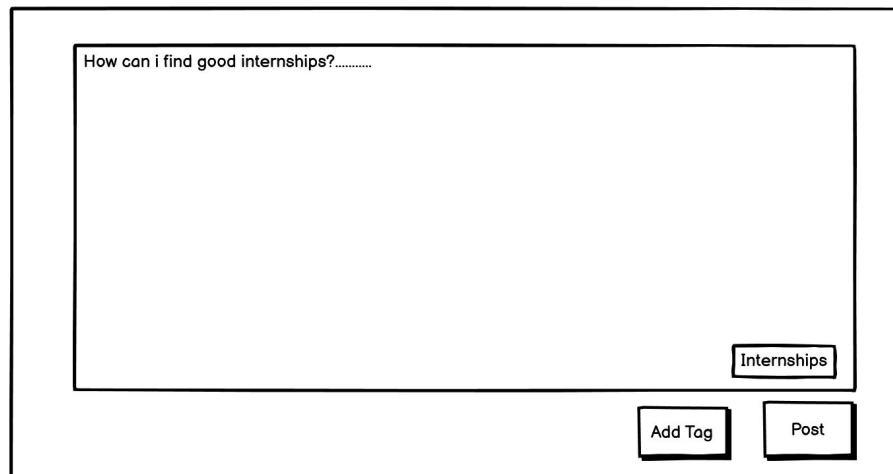
...

Student forum page is a section of the application which is independent from courses' pages. It can be reached by "My Courses Page" and it is linked to that page. This page shows questions that are asked in the forum. Post button is linked to "Create Post Page" and also when clicked on a question, the user is directed to "Show Post Page". Also the user may search posts by keywords.

3.5.2.19 Create Post Page

[← Back to my course](#)

Student Forum



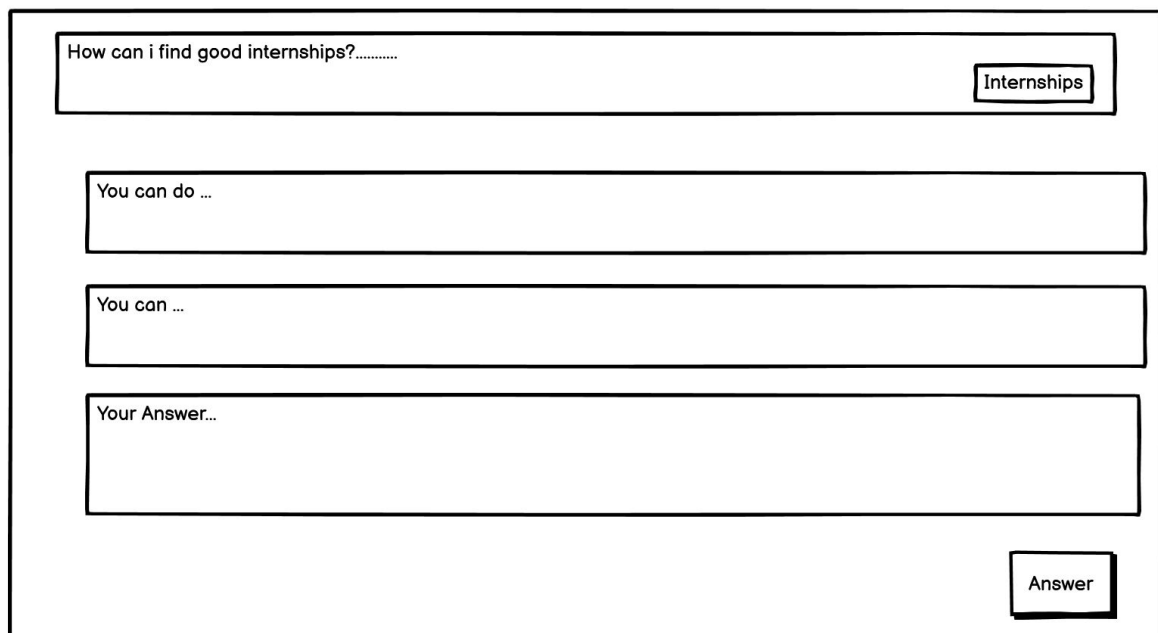
The mockup shows a large text input area for writing a post. Inside the input area, the text "How can i find good internships?....." is visible. To the right of the input area, there is a small button labeled "Internships". Below the input area, there are two buttons: "Add Tag" and "Post".

This Page allows the user to create a post and write the question of the post.

3.5.2.20 Show Post Page

[← Back to my course](#)

Student Forum



The mockup shows a post detail view. At the top, there is a text input area containing the question "How can i find good internships?....." and a button labeled "Internships". Below this, there are three text input areas for answers, each with a placeholder text: "You can do ...", "You can ...", and "Your Answer...". At the bottom right, there is a button labeled "Answer".

This page shows the question of a forum post and its answers. User can write his/her answer using the text field and post it by clicking the Answer button.

4. Glossary & References

Resources used in the sequence diagram:

<http://www.cs.sjsu.edu/~pearce/oom/ood/distArch/server>

https://www.researchgate.net/figure/UML-collaboration-diagram-of-the-client-server-architecture_fig1_30530658

UI Designs are inspired by these web sites (login required to see applications):

<https://mlm.pearson.com/northamerica/masteringphysics/>

<https://moodle.bilkent.edu.tr/2020-2021-spring/login/index.php>

Designs made through Balsamiq Wireframes:

<https://balsamiq.com/>