CS319 D1

Pool

T1-havuz

Abdurrahman Bilal Kar 22003569

Ahmet Memduh Tutuş 22003153

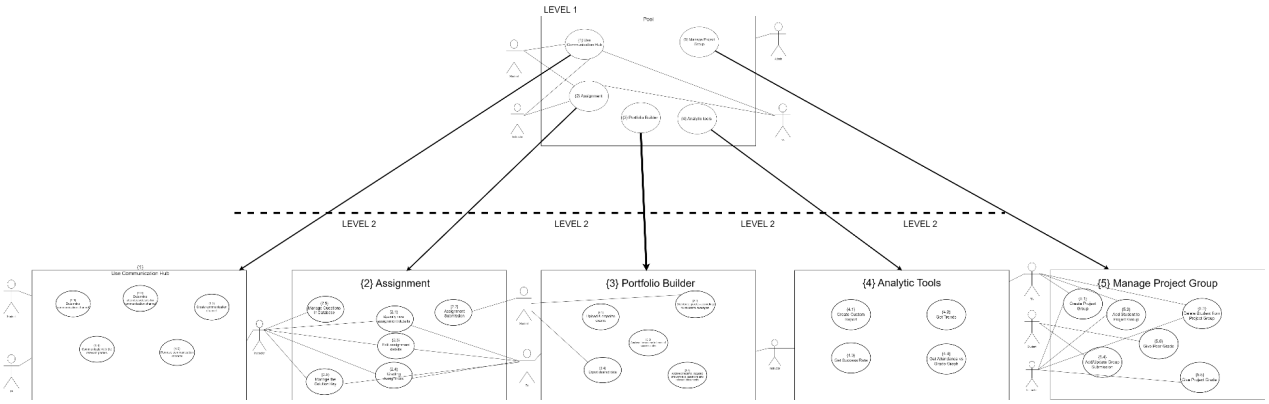
Arda Kırcı 22002031

Berkay Ayçiçek 22003111

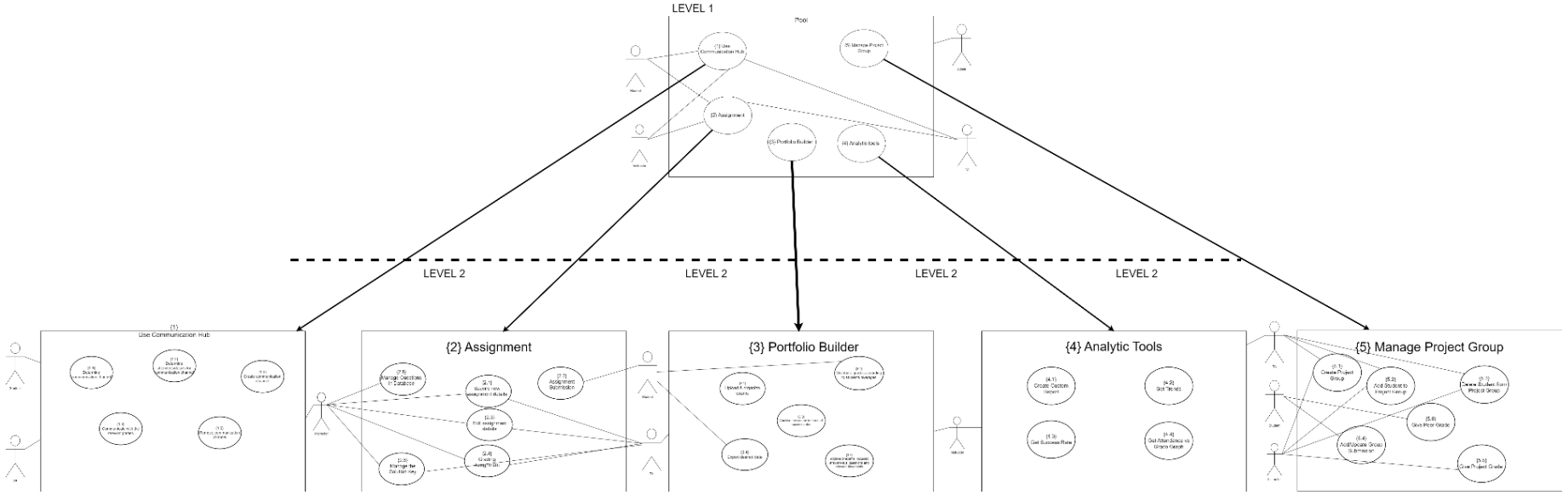
Fırat Utku Gül 22003105

25.02.2024

1. Use Case Diagram



To see clearly, the user case diagram is full on the next page, in moodle, everyone submitted their own parts.


LEVEL 1
Pool
(1) Use Communication Hub
(5) Manage Project Group
(2) Assignment
(3) Portfolio Builder
(4) Analytic tools
LEVEL 2
LEVEL 2
LEVEL 2
LEVEL 2
{1} Use Communication Hub
{2} Assignment
{3} Portfolio Builder
{4} Analytic Tools
{5} Manage Project Group
(7.5) Assignment Submission
(4.1) Create Custom Report
(4.2) Set Thresholds
(4.3) Get Success Rate
(4.4) Get Attendance vs Grade Graph
(5.1) Create Project Group
(5.2) Add Students to Project Group
(5.3) Delete Student from Project Group
(5.4) Add/Update Group Submission
(5.5) Give Project Grade
(5.6) Give Peer Grade

2. Use Case Narratives

2.1. Use of Communication Hub - Bilal

Communication hub is the place where students, instructors and teaching assistants can create forums, post forum posts, create group messages, and create events. There are 5 use cases in it.

2.1.1. Determine Communication Channel

Actors:

- Student
- Instructor
- TA

Description:
The Determine Communication Channel use case involves the actors (Student, Instructor, or TA) deciding whether to utilize a forum, a chat, or an event within the course management system based on their communication needs.

Preconditions:

- The course management system is operational.
- The actor has a valid account.

Basic Flow:

- The actor accesses the course management system and identifies the need for communication.
- The system evaluates the communication requirements and options based on the actor's intent.
- If no issues are detected, the system allows the actor to proceed with the chosen communication method.
- The system updates the communication channel based on the actor's decision.

Postconditions:

- The communication channel is determined based on the actor's decision.
- If adjustments are made, the system reflects the updated communication method.

Exceptional Flow:

- If technical difficulties arise during the selection process, the system generates an error message.

- The actor reports the issue to the technical support team for resolution, if necessary.

2.1.2. Determine Attendees/Details for Communication Channel

Actors:
- Student
- Instructor
- TA

Description:
The Determine Attendees/Details for Communication Channel use case involves the actors (Student, Instructor, or TA) deciding on the participants and specific details for communication within the course management system, such as selecting recipients for messages or setting event parameters such as date, place etc.

Preconditions:
- The course management system is operational.
- The actor has a valid account.
- Communication channel has been selected by the actor.

Basic Flow:
- The actor accesses the course management system and identifies the need for communication with specific attendees or regarding specific details.
- The system evaluates the communication requirements and options based on the actor's intent.
- If no issues are detected, the system allows the actor to proceed with selecting attendees and specifying communication details.
- If constraints or limitations are identified (e.g., insufficient permissions, conflicting schedules), the system prompts the actor accordingly.
- The actor reviews the notification and adjusts the selected attendees or details if necessary.
- The system updates the communication parameters based on the actor's decision.

Postconditions:
- The communication channel's attendees and details are determined based on the actor's decision.

- If adjustments are made, the system reflects the updated communication parameters.

Exceptional Flow:

- If technical difficulties arise during the selection process, such as the number of participants for a certain communication channel, the system generates an error message, along with the capacity constraints of the channel selected.

2.1.3. Create Communication Channel

Actors:

- Student
- Instructor
- TA

Description:
The Create Communication Channel use case involves the actors (Student, Instructor, or TA) initiating the creation of a new communication channel within the course management system to facilitate interaction, collaboration, or information sharing.

Preconditions:

- The course management system is operational.
- The actor has a valid account.
- The communication type has been selected.
- The participants and the necessary details are filled.

Basic Flow:

- The actor accesses the course management system and identifies the need for communication.
- The system evaluates the communication requirements and options based on the actor's intent.
- If no issues are detected, the system allows the actor to create a channel for the chosen communication method.
- If there are constraints or limitations identified (e.g., system overload, inappropriate channel for the intended message), the system prompts the actor accordingly.
- The system creates the communication channel based on the actor's decision.

Alternate Flow: Inappropriate Choice of the Channel:

- If the chosen communication channel is inappropriate (e.g., if sensitive information needs to be discussed), the actor selects an alternative method or platform.

Postconditions:
- The communication channel is determined based on the actor's decision.
- If adjustments are made, the system reflects the updated communication method.
- All relevant participants are informed of the chosen communication channel and any necessary details or instructions.

Exceptional Flow:
- If technical difficulties arise during the selection process, the system generates an error message.
- The actor reports the issue to the technical support team for resolution, if necessary.

2.1.4. Communicate with the Relevant Parties

Actors:
- Student
- Instructor
- TA

Description:
The Communicate with Relevant Parties use case involves the actors (Student, Instructor, or TA) engaging in communication with other relevant parties within the course management system. This communication can occur through various channels such as forums, chats, or events, depending on the nature of the interaction.

Preconditions:
- The course management system is operational.
- The actor has a valid account.
- A communication channel has been created or selected.

Basic Flow:
- The actor navigates to the chosen communication channel within the system.
- The actor composes and sends the message, providing necessary details, instructions, or information to the relevant parties.

- The system delivers the message to the intended recipients within the selected communication channel.
- Recipients receive the message and can engage in further communication or actions as necessary.

Alternate Flow: Communication with Participants:

- The actor communicates respectfully and clearly with other participants, providing necessary details or instructions.
- If the chosen communication channel is inappropriate (e.g., if sensitive information needs to be discussed), the actor selects an alternative method or platform.
- In case of miscommunication or misunderstanding, the actor clarifies the message promptly.

Postconditions:

- If adjustments are made, the system reflects the updated communication method.
- All relevant participants are informed of the chosen communication channel and any necessary details or instructions.

Exceptional Flow:

- If technical difficulties arise during the selection process, the system generates an error message.
- The actor reports the issue to the technical support team for resolution, if necessary.
-

2.1.5. Remove Communication Channel

Actors:

- Student
- Instructor
- TA

Description:

The Remove Communication Channel use case involves the actors (Student, Instructor, or TA) initiating the removal of an existing communication channel within the course management system, effectively discontinuing its usage and access.

Preconditions:

- The course management system is operational.

- The actor has a valid account.
- The actor has the authority to remove the channel.
- The targeted communication channel exists within the system.

Basic Flow:

- The actor accesses the course management system and navigates to the communication tools section or the list of existing channels.
- The actor identifies the communication channel to be removed.
- The actor selects the option to delete or remove the channel.
- The system prompts the actor to confirm the deletion of the selected channel.
- The actor confirms the deletion request.
- The system removes the communication channel from the list of active channels and revokes access for all participants.

Postconditions:

- The targeted communication channel is successfully removed from the course management system.
- Access to the channel is revoked for all participants, and it is no longer accessible.
- Any associated data or content within the removed channel may be archived or deleted as per system policies.

Exceptional Flow:

- If technical difficulties arise during the selection process, the system generates an error message.
- The actor reports the issue to the technical support team for resolution, if necessary.

2.2. Portfolio Builder - Memduh

2.2.1. Upload & Organize Exams

Actors:

- Instructor

- TA

Description:
To help instructors and TA's to organize exams in terms of their questions, the style, points, and also helps instructors to upload the exams in a pdf format. This attribute works along with export data.

Preconditions:
- The course management system is operational.
- The actor has a valid account or is logged in.
- Exam's course is selected.
- For upload, the type of document should be PDF.
- For organization, actors should have prior questions to load in, or should create questions via using a question database.

Basic Flow:
- The actor accesses the course management system and uploads the document.
- The system evaluates whether the type of document is accurate.
- If no issues are detected, the system allows the actor to proceed with two selections, organize the questions or create new ones.
- If organization is selected, the actor should organize the document in terms of question numbers, points, and total number of questions.
- If create mode is selected, the actor should add a new question to that exam. If there are no empty values in the created question, this question is added to the exam.
- Later, the actor may download the file as likes.

Postconditions:
- The upload and organization system is determined based on the actor's decision.
- If adjustments are made, the system will save the last version of that exam. Actors should be sure that this is what is wanted.

Exceptional Flow:
- If technical difficulties arise during the selection process, the system generates an error message.
- The actor reports the issue to the technical support team for resolution, if necessary.

2.2.2. Distributing Grades

Actors:

- Instructor
- TA
- Student

Description:

To assist TA's and instructors to enter student's grades into the system. This feature also helps to see what the average of the exam is, and according to the instructor's choice, letter grades are given to students.

Preconditions:

- The course management system is operational.
- The actor has a valid account or is logged in.
- The exam should already be made, and graded by the instructor.
- There should be students in the chosen section, otherwise there is no point.
- The corresponding letter grade for that average and points to upgrade the letter grade should be decided by the instructor.

Basic Flow:

- The actor accesses the course management system and enters the evaluation page.
- The user asks for the section that is wanted to be evaluated.
- If no issues are detected, the system allows the actor to proceed with entering the grades.
- After entering all student's grades, the instructor should decide letter grade for the average, and upgrade points for letter grades.
- If there are no issues, the instructor must press the publish button.
- Later, the actor may edit the notes as likes.
- Students can see their grades after each midterm through selecting exam results.

Alternate Flow: Objection and re-evaluate with students:

- The actor communicates by using a communication hub, then arranges a time interval for the objection session.
- If there are miscalculations noticed in the objection session, the actor(instructor or TA) can re-enter the student's correct note.

Postconditions:

- The upload and organization system is determined based on the actor's decision.
- If adjustments are made, the system will save the last version of that exam's results.
- The actor should be certain about pressing the publish button.

Exceptional Flow:

- If technical difficulties arise during the selection process, the system generates an error message.
- The actor reports the issue to the technical support team for resolution, if necessary.
- If a student withdraws from the course, grades that he got will not affect the average of the exam, or distribution of the grades.

2.2.3. Analyzing the Course

Actors:

- Instructor
- TA

Description:
To see past semesters, their note averages, the correlation between attendance and received note in a good-looking table, and track the instructors average section points.

Preconditions:

- The course management system is operational.
- The actor has a valid account or is logged in.
- The semester should already be finished
- All grades and notes should be finalized.
- Analyzed course’s term (spring-fall-year) should be selected.

Basic Flow:

- The actor accesses the course management system and enters the analyze page.
- The user asks for the course and the term that is wanted to be analyzed.
- If no issues are detected, the system allows the actor to proceed with seeing the analysis of the grades, provided with names, attendance records, and ID's of the students.
- After entering required attributes, the instructor can see the plot of analysis and relations.
- If there are no issues, the instructor can export plots.

Postconditions:

- With the help of analyzing tools, instructors can see relations between attendance and success.
- By combining multiple terms of the given course, further analysis can be made.

Exceptional Flow:

- If technical difficulties arise during the selection process, the system generates an error message.
- The actor reports the issue to the technical support team for resolution, if necessary.
- If a student withdraws from the course, grades that he got will not affect the analysis of the exam

2.2.4. Export Data

Actors:

- Instructor
- TA
- Student

Description:

To export wanted data, such as plots for analysis, attendance tracks, transcript and so on.

Preconditions:

- The course management system is operational.
- The actor has a valid account or is logged in.
- There should already be some data.
- If the actor is a student, then he cannot access analysis plots and previous exams.

Basic Flow:

- The actor accesses the course management system and enters the page that he wanted to export.
- The user presses the button to download which track is available in that page.
- If no issues are detected, the system allows the actor to proceed with downloading the data as a table.
- Later, the actor may download the same file as likes.
- Students can see their transcripts after a full semester. Beforehand, it is not available for that student.

Postconditions:

- The actor will have a PDF file with the context of whatever they downloaded.
- If the actor wants to re-download the same data, it will be seen as NAME_OF_DATA(2).pdf and so on.

Exceptional Flow:

- If technical difficulties arise during the selection process, the system generates an error message.
- The actor reports the issue to the technical support team for resolution, if necessary.
- If a student has withdrawn from the course, then the analysis of attendance, grades, and this semester's course page will be hidden for that.

2.2.5. Archive of Exams

Actors:

- Instructor
- TA

Description:

To access previous semester's results, and compare all together in terms of attendance ratio, average results, and success of sections in each exam.

Preconditions:

- The course management system is operational.
- The actor has a valid account or is logged in.
- Previous term's results must be provided by the instructors.
- There should be at least one section for a course.

- All the required fields must be filled by the instructor.

Basic Flow:

- The actor accesses the course management system and enters the evaluation page.
- The user asks for the section that is wanted to be evaluated.
- If no issues are detected, the system allows the actor to proceed with viewing previous exams.
- If the actor wants to export a plot, or a table, then the wanted table must be specified.

Alternate Flow: Missing Arguments:

- If there are some missing arguments in previous exams somehow, the table included that arguments will be seen as NULL.

Postconditions:

- Comparison between sections, and previous semester's sections, and all together semesters can be made.
- With new semesters, the archive will be expanded.

Exceptional Flow:

- If technical difficulties arise during the selection process, the system generates an error message.
- The actor reports the issue to the technical support team for resolution, if necessary.

2.3. Assignment Builder - Fırat

2.3.1. Specify new Assignment Details

Actors:

- Instructor
- TA

Description:

The Specify new Assignment Details use case involves the actors (Instructor or TA) creating a new assignment by choosing the assignment details, effectively creating the new assignment for the students taking respective course.

Preconditions:

- The course management system is operational.
- The actor has a valid account or is logged in.
- Assignment's course is selected.
- For creation, the actor should fill all the details of the assignment.

Basic Flow:

- The actor accesses the course management system.
- The actor fills the details of the assignment.
- If every detail is filled, the system allows the actor to create the new assignment.
- If not every detail is filled, the system directs you to Specify new Assignment Details use case again.
- Later, the actor may download the assignment as PDF.

Postconditions:

- The system maintains a record of the assignment in the actor's account history.

Exceptional Flow:

- If technical difficulties arise during the selection process, the system generates an error message.
- The actor reports the issue to the technical support team for resolution, if necessary.

2.3.2. Assignment Submission

Actor:

- Student

Description:
The Assignment Submission use case involves students submitting their answers to the created assignment.

Preconditions:

- The course management system is operational.
- The actor has a valid account or is logged in.
- Submission's course is selected.
- Assignment should be previously created.
- For submission, the actor should upload a correct file type defined by the instructor/TA.

Basic Flow:

- The actor accesses the course management system.
- The actor submits their assignment.
- If the submission is in the correct file type, the system accepts the submission.
- If the submission is in the wrong file type, the system doesn't accept the submission and gives an error message.

Alternate Flow: Passing the due date:

- If the assignment's due date is passed, the system doesn't accept the submission, and gives the actor to connect to instructor/TA about the issue.

Postconditions:

- The system maintains a record of the submission in the actor's account history.

Exceptional Flow:

- If technical difficulties arise during the selection process, the system generates an error message.
- The actor reports the issue to the technical support team for resolution, if necessary.

2.3.3. Edit Assignment Details

Actors:

- Instructor
- TA

Description:
The Edit Assignment Details use case involves the actors (Instructor or TA) editing an assignment, editing the assignment for the students taking their respective course.

Preconditions:

- The course management system is operational.
- The actor has a valid account or is logged in.
- Assignment's course is selected.
- To edit an assignment, it should be previously created.

Basic Flow:

- The actor accesses the course management system.

- The actor selects the assignment to edit.
- If every detail is filled, the system allows the actor to edit the assignment.
- If not every detail is filled, the system gives an error and halts the editing process, without changing the initial assignment.
- Later, the actor may download the assignment as PDF.

Alternate Flow: Trying to edit an outdated assignment:

- If the assignment's due date is passed, the system doesn't accept the edition, and throws an error.

Postconditions:

- The system maintains a record of the assignment in the actor's account history.

Exceptional Flow:

- If technical difficulties arise during the selection process, the system generates an error message.
- The actor reports the issue to the technical support team for resolution, if necessary.

2.3.4. Grading Assignment

Actors:

- Instructor
- TA

Description:

The Grading Assignment use case involves the actors (Instructor or TA) grading an assignment, grading the assignment for the students taking their respective course.

Preconditions:

- The course management system is operational.
- The actor has a valid account or is logged in.
- Assignment's course is selected.
- To grade an assignment, it should be previously created.
- To grade an assignment, its due date should have passed.

Basic Flow:

- The actor accesses the course management system.
- The actor selects the assignment to grade.

- If assignment is suitable to grade, system allows actor to grade the assignment.

Alternate Flow: Trying to edit an outdated assignment:

- If the assignment's due date has not passed, the system doesn't accept grading, and throws an error.

Postconditions:

- The system maintains a record of the assignment in the actor's account history.

Exceptional Flow:

- If technical difficulties arise during the selection process, the system generates an error message.
- The actor reports the issue to the technical support team for resolution, if necessary.

2.3.5. Manage Questions in Database

Actors:

- Instructor

Description:
The Manage Question in Database use case involves the instructor managing questions in the database, creating, editing, or deleting questions in the database.

Preconditions:

- The course management system is operational.
- The actor has a valid account or is logged in.
- To edit a question, it should be previously created.
- To delete a question, it should be previously created.

Basic Flow:

- The actor accesses the course management system.
- The actor selects the option to create, edit, or delete a question.
- If every detail is filled, the system allows the actor to create a question.
- If every detail is not filled, the system doesn't allow the actor to delete a question.
- If every detail is filled, the system allows the actor to edit the question.
- If not every detail is filled, the system gives an error and halts the editing process without changing the initial question.
- The actor can delete questions from the database.

Postconditions:

- The system maintains a record of the questions in the question history.

Exceptional Flow:

- If technical difficulties arise during the selection process, the system generates an error message.
- The actor reports the issue to the technical support team for resolution, if necessary.

2.3.6. Manage the Solution Key

Actors:

- Instructor
- TA

Description:
The Manage Solution Key use case involves the instructor managing the solution key for a particular assignment, creating or editing the solution key for the grading process of the assignment.

Preconditions:

- The course management system is operational.
- The actor has a valid account or is logged in.
- To edit a solution key, it should be previously created.

Basic Flow:

- The actor accesses the course management system.
- Assignment's course is selected.
- Assignment is selected.
- To create the solution key, actor uploads a pdf file.
- To edit the solution key, actor uploads a pdf file.
- If the uploaded file is not a pdf system doesnt allow the actor to create or edit the solution key and halts the process with no changes.

Postconditions:

- The system maintains a record of the solution keys in the assignment history.

Exceptional Flow:

- If technical difficulties arise during the selection process, the system generates an error message.
- The actor reports the issue to the technical support team for resolution, if necessary.

2.4. Analytic Tools - Arda

2.4.1. Create Custom Report

Actors:

- TA
- Instructor

Description:
The Create Custom Report use case involves the Instructor or a TA to create a custom report based on a filter, or some criteria that is specified by the actors.

Preconditions:

- The course management system is operational.
- The actor has a valid account and is logged in.
- The related course, assignment, student is selected for the report.
- The actor initiating the creation of the custom report has appropriate permissions.

Basic Flow:

- The actor accesses the course management system and navigates to the report creation page.
- The actor selects the desired filters to create a report.
- If the filters correspond to some data, then it appears as a report.
- If the actor does not specify any filter, the report then will not be seen, or will be seen as all the data included in that category.

Postconditions:

- The report that is customized by the actor is available at the end of the filtering process.
- That report can be used in further applications, or examinations.

Exceptional Flow:

- If technical difficulties arise during the selection process, the system generates an error message.
- The actor reports the issue to the technical support team for resolution, if necessary.

2.4.2. Get Trends

Actors:
- TA
- Instructor

Description:
The Getting Trends use case involves the Instructor or a TA to create a specific filtering designed for a specific student. By that, instructors or TA's can track a student's past semester grades, attendance ratios for some courses.

Preconditions:
- The course management system is operational.
- The actor has a valid account and is logged in.
- The related student is selected by the actors.
- Actors can filter which attributes will be seen about the student.

Basic Flow:
- The actor accesses the course management system and navigates to the trend identification page.
- The actor selects the desired filters and the student.
- If the filters correspond to some data, then it appears as a report.
- If that student does not exist in the database, then the actor will get an error that tells the student it did not exist previously.
- If the actor does not specify any filter, the report then will show all the semesters, attendance ratios, and grades of that student.

Postconditions:
- The report that is customized by the actor is available at the end of the filtering process. And it can be re-used whenever wanted.
- After that student's graduation, his/her records will be kept in the system.

Exceptional Flow:

- If technical difficulties arise during the selection process, the system generates an error message.
- The actor reports the issue to the technical support team for resolution, if necessary.

2.4.3. Get Success Rate

Actors:

- TA
- Instructor

Description:
The Success Rate use case involves the Instructor or a TA to create a specific filtering designed for questions. By that, instructors or TA's can track exams, or even a question's toughness, numbers of students that are solved that question, and average points that are taken from that question.

Preconditions:

- The course management system is operational.
- The actor has a valid account and is logged in.
- The related question or exam is selected by the actors.
- The actors should select the course, the semester, and if they wanted, the questions.

Basic Flow:

- The actor accesses the course management system and navigates to the success rate analysis page.
- The actor selects the desired filters and the question.
- If the filters correspond to some data, then it appears as a report.
- If that question does not exist in the database or makes some input mistakes, then the actor will get an error.
- By doing that, the actors will have a clean report that explicitly shows in which topics that students are struggling, and how many tough, middle and easy questions that should be in the exam.
- If the actor does not specify any filter, the report then will show all the exams and questions.

Postconditions:

- The report that is customized by the actor is available at the end of the filtering process, and it can be re-used whenever wanted.
- At the end of each semester, that semester's question will be added to the database.

Exceptional Flow:
- If technical difficulties arise during the selection process, the system generates an error message.
- The actor reports the issue to the technical support team for resolution, if necessary.

2.4.4. Get Attendance/Grade Graph

Actors:
- Instructor
- TA

Description:
Get Attendance/Grade Graph use case involves the Instructor or a TA getting the ratio of Attendance and Grade as a graph within the course management system.

Preconditions:
- The course management system is operational.
- The actor has a valid account.
- The related course and student(s) is selected.

Basic Flow:
- The actor accesses the course management system and navigates to get attendance/grade graph section.
- The actor specifies the options to create the graph.
- The system prompts the user to provide necessary details for the graph.
- The new graph is successfully created within the course management system.

Postconditions:
- The system maintains the graphs in the system's history.

Exceptional Flow:

- If technical difficulties arise during the selection process, the system generates an error message.
- The actor reports the issue to the technical support team for resolution, if necessary.

.

2.5. Create Project Groups - Berkay

2.5.1. Create Project Group

Actors:

- Instructor
- TA

Description:
The Create Project Group use case involves the actor initiating the creation of a new project group within the course management system. This action enables students to collaborate on assignments or projects as a unit.

Preconditions:

- The course management system is operational.
- The actor has a valid account.
- The related course and assignment for the project group is selected.
- Actor initiating the creation of the project group has appropriate permissions.

Basic Flow:

- The actor accesses the course management system and navigates to the project group creation section.
- The actor selects the option to create a new project group.
- The system prompts the user to provide necessary details for the new group, such as the group name, description, and group members.
- The actor enters the required information, including the names or IDs of group members.
- The system validates the provided information and creates the new project group accordingly.
- The system assigns a unique identifier or code to the newly created project group.
- The actor receives a confirmation message indicating the successful creation of the project group.

Postconditions:

- The new project group is successfully created within the course management system.
- Group members are added to the project group and granted appropriate access rights.
- The project group is available for collaboration, discussions, and submission of group assignments or projects.

Exceptional Flow:

- If technical difficulties arise during the selection process, the system generates an error message.
- The actor reports the issue to the technical support team for resolution, if necessary.

2.5.2. Add Student to Project Group

Actors:

- Instructor
- TA

Description:
The Add Student to Project Group use case involves the appropriate actor adding a student to an existing project group within the course management system. This action enables students to collaborate effectively on group assignments or projects.

Preconditions:

- The course management system is operational.
- The actor has a valid account.
- Actor initiating the creation of the project group has appropriate permissions.
- The Student to be added is enrolled in the course.

Basic Flow:

- The actor accesses the course management system and navigates to the project group management section.
- The actor selects the specific project group to which the Student will be added.
- The actor identifies and selects the option to add a Student to the group.

- The system prompts the actor to input the necessary details of the Student to be added, such as their name or student ID.
- The actor enters the required information for the Student and confirms the addition.
- The system verifies the Student's details and adds them to the selected project group.
- The Student receives a notification indicating their successful addition to the project group.

Postconditions:
- The Student is successfully added to the specified project group within the course management system.
- The Student gains access to group resources, discussions, and collaboration tools associated with the project group.
- The group members are notified of the Student's addition to the group.

Exceptional Flow:
- If technical difficulties arise during the selection process, the system generates an error message.
- The actor reports the issue to the technical support team for resolution, if necessary.

2.5.3. Delete Student from Project Group

Actors:
- Instructor
- TA

Description:
The Delete Student from Project Group use case involves the appropriate actor removing a student from an existing project group within the course management system. This action facilitates the management of project groups and ensures that only relevant students are participating in group assignments or projects.

Preconditions:

- The course management system is operational.
- The actor has a valid account.
- Actor initiating the creation of the project group has appropriate permissions.
- The student to be removed is currently a member of the project group.

Basic Flow:

- The actor accesses the course management system and navigates to the project group management section.
- The actor selects the specific project group from which the student will be removed.
- The actor identifies and selects the option to delete a student from the group.
- The system presents a list of current group members for the selected project group.
- The actor selects the student to be removed from the group.
- The actor confirms the deletion of the selected student.
- The system removes the student from the project group membership.
- The student receives a notification indicating their removal from the project group.

Postconditions:

- The group submission is successfully added/updated for the specified project group within the course management system.
- The submission is available for review, evaluation, and grading.
- The Student receives a confirmation of the successful submission/update.

Exceptional Flow:

- If technical difficulties arise during the selection process, the system generates an error message.
- The actor reports the issue to the technical support team for resolution, if necessary.

2.5.4. Add/Update Group Submission

Actors:

- Student

Description:

The Add/Update Group Submission use case involves a student making or updating a submission on behalf of their project group within the course management system. This action facilitates the submission of group assignments or projects and ensures accurate assessment and evaluation.

Preconditions:

- The course management system is operational.
- The project group has been created and assigned a submission task.
- The student initiating the submission/update has appropriate permissions.

Basic Flow:

- The Student accesses the course management system and navigates to the submission section for the project group.
- The Student selects the specific project group for which the submission will be added or updated.
- The Student identifies and selects the option to add/update the group submission.
- The system prompts the Student to upload the necessary files or documents for the group submission.
- The Student uploads the required files or documents and provides any additional information or comments related to the submission.
- The system verifies the submission files and details, associating them with the selected project group.
- The submission is successfully added/updated in the course management system, and a confirmation message is displayed to the Student.

Postconditions:

- The group submission is successfully added/updated for the specified project group within the course management system.
- The submission is available for review, evaluation, and grading.
- The members of the group receive a confirmation of the successful submission/update.

Exceptional Flow:

- If technical difficulties arise during the selection process, the system generates an error message.
- The actor reports the issue to the technical support team for resolution, if necessary.

2.5.5. Give Project Grade

Actors:

- Instructor
- TA

Description:

The Give Project Grade use case involves Instructor or TA assigning a grade and a feedback to a project submitted by a project group within the course management system. This action facilitates the evaluation and assessment of group assignments or projects.

Preconditions:

- The course management system is operational.
- The project group has submitted their project for grading.
- The Instructor or TA has appropriate permissions to grade projects.

Basic Flow:

- The actor accesses the course management system and navigates to the grading section for the project.
- The actor selects the specific project submitted by the project group.
- The actor identifies and selects the option to give a project grade.
- The system presents a grading form or interface, allowing the actor to input the grade and provide feedback/comments if required.
- The actor assigns the grade and submits the project evaluation.
- The system records the grade and feedback provided by the actor for the project.

Postconditions:

- The project submitted by the project group is successfully graded by the Instructor.
- The grade and feedback provided by the Instructor are recorded in the course management system.
- The project group members can view their grade and feedback for the project.

Exceptional Flow:

- If technical difficulties arise during the selection process, the system generates an error message.
- The actor reports the issue to the technical support team for resolution, if necessary.

2.5.6. Give Peer Grade

Actors:
- Student

Description:
The Give Peer Grade use case involves a student assigning a grade to a peer's work within a project group, promoting peer evaluation and feedback in the course management system.

Preconditions:
- The course management system is operational.
- Peer grading functionality is enabled for the assignment or project.
- The student providing the peer grade is a member of the project group.

Basic Flow:
- The Student accesses the course management system and navigates to the peer grading section for the assignment or project.
- The Student selects the specific submission or peer work for which the grade will be given.
- The Student identifies and selects the option to give a peer grade.
- The system presents a grading form or interface, allowing the Student to input the grade and provide feedback/comments if required.
- The Student assigns the grade and submits the peer evaluation.
- The system records the peer grade and feedback provided by the Student.

Postconditions:
- The peer grade is successfully assigned to the peer's work within the project group.
- The graded student receives feedback and the assigned grade for their work.
- The peer grading process contributes to the overall assessment and evaluation of the assignment or project.

Exceptional Flow:

- The actor reports the issue to the technical support team for resolution, if necessary.

3. Non-Functional Requirements

3.1. Performance

- Response Time: The application is expected to be highly responsive, with a target response time of under 5 seconds for most user interactions, ensuring a smooth user experience.
- Scalability: The system must possess robust scalability features to effortlessly accommodate the increasing number of users and items within the marketplace as it grows.
- Load Handling: It is imperative for the application to efficiently manage heavy loads, capable of seamlessly handling a large influx of concurrent users without experiencing performance degradation.

3.2. Security

- Security: Stringent measures will be employed to ensure the utmost security of user data, particularly personal and financial information, utilizing advanced encryption protocols and reliable database systems like MongoDB.
- Authentication: Secure authentication mechanisms will be implemented, allowing only authorized Bilkent University users through their official email addresses. Unauthorized access attempts from external entities will be strictly prohibited.
- Data Privacy: User privacy and data protection will be treated as paramount concerns, with strict adherence to relevant data protection regulations to safeguard user confidentiality.

3.3. Reliability

- Availability: Uninterrupted availability is crucial, with the system expected to be operational 24/7, ensuring users have constant access to its functionalities with minimal downtime for maintenance.
- Data Integrity: The integrity and reliability of user data, listings, and messages will be ensured through regular database backups and robust storage mechanisms, mitigating the risk of data loss or corruption.

3.4. Maintainability:

- Object-Oriented Programming principles will guide the development process, promoting modular and extensible code structures to facilitate future enhancements and feature additions.
- Comprehensive documentation and adherence to coding standards will streamline maintenance efforts, making it easier for developers to navigate and modify the codebase as needed.
- These practices will also expedite the debugging process, enabling swift identification and resolution of any issues that may arise.

3.5. Usability

- The application's interface will prioritize user-friendliness and visual appeal, designed to provide users with an intuitive and pleasant experience. Each feature will be clearly delineated and accompanied by concise, informative descriptions to enhance usability.

4. Technology Stack

The technology stack chosen for the project consists of Flutter for the frontend, Node.js for the backend, and MongoDB for the database.

4.1. Frontend

Flutter was chosen for frontend development because of its cross-platform functionality, which allows for the development of both Android and iOS applications from a single codebase. Its broad widget library and hot reload functionality simplify the programming process, increasing developer productivity and allowing for rapid iteration.

4.2. Backend

Node.js was chosen for backend development because of its event-driven, non-blocking I/O style, which allows it to efficiently handle concurrent queries. Its package ecosystem, together with frameworks such as Express.js, offers powerful tools for developing scalable and performant server-side applications. Furthermore,

Node.js complements Flutter for full-stack JavaScript development, allowing for easy integration of frontend and backend components.

4.3. Database

MongoDB acts as the project's database solution, providing a flexible and scalable NoSQL database that stores data as JSON-like documents. Its schema-free architecture allows for flexible data structures and quick iterations throughout development. MongoDB's distributed architecture and support for horizontal scaling assure consistent speed and availability, making it ideal for modern web and mobile applications.