



Bilkent University

Department of Computer Engineering

CS319 Term Project

Section 3

Group 3A

Bilasmus

Analysis Report

Group Members:

Alp Afyonluoğlu (22003229)
Emre Karataş (22001641)
Erkin Aydın (22002956)
Parsa Keihan (22001422)
Selin Bahar Gündoğar (22001514)

Instructor: Eray Tüzün

Teaching Assistant: Muhammad Umair Ahmed

1. Contents

1.	Introduction	3
2.	Proposed System	3
2.1.	Overview	3
2.2.	Functional Requirements	4
2.3.	Non-functional Requirements	6
3.	System Models	7
3.1.	Use-Case Diagram	7
3.2.	Class Diagram	19
3.3.	Dynamic Models Diagram	22
3.3.1.	State Machine Diagrams	22
3.3.1.1.	Learning Agreement Upload	22
3.3.1.2.	Course Request Process	22
3.3.2.	Activity Diagrams	23
3.3.2.1.	Course Transfer Form Evaluation	23
3.3.2.2.	Course Identification	23
3.3.3.	Sequence Diagrams	24
3.3.3.1.	Course Requesting	24
3.4.	User Interface	24
3.5.	Improvement Summary	41

Analysis Report

Bilasmus

1. Introduction

Our application Bilasmus is a mode to communicate the Erasmus/Exchange mobility processes of outgoing Bilkent students and incoming students from partner universities with Bilkent Erasmus/Exchange coordinators and personnel. The previous mode of carrying out this procedure included all parties of Erasmus/Exchange communicating through emails and handing paperwork by hand. Thus, as the members of Erasmus, we decided to create an application that would help all people associated with the Erasmus/Exchange process.

This application will contribute to the efficiency of processing official and non-official Erasmus documents within Bilkent. The application will eradicate the paper-based communication and will minimize the need to send emails to related users. Thus, as the group, Bilasmus, we created this application to be used by Erasmus/Exchange coordinators, incoming and outgoing students, Faculty Committee Board, International Student Office, and Department Secretaries.

All users will be able to upload the necessary documents they need to upload as for their role and e-sign the documents through the application as well. The users will be getting notifications both on the application and through email to be reminded of other related users' progress. Furthermore, each user will be having a to-do list on the application. All users will be able to send messages to related users. Information of all users will be given on their profile pages. Moreover, the students can also send their course request forms through the application to both instructors, coordinators, and department secretaries.

2. Proposed System

The website application will manage the process of submitting and reviewing mobility documents related to Erasmus/Exchange of outgoing Bilkent students and incoming students.

2.1. Overview

The website application will manage the process of submitting and reviewing mobility documents related to Erasmus/Exchange of outgoing Bilkent students and incoming students.

The application will be serving all Erasmus users by offering different user type logins to the application. The outgoing students will be able to upload their necessary documents such as the preapproval form, language proficiency exam results, and course request forms. Moreover, outgoing student users can also see a list of all previously accepted courses before creating their course request forms. The incoming students will be able to send their course requests to the department secretary. Additionally, all Erasmus/Exchange coordinators will be able to review the documents uploaded by the outgoing students and e-sign it or reject the documents and request change from the students. Followingly, the coordinator can also upload the course transfer form through the application to the Faculty Committee Board for the approval of such document. The Faculty Committee Board will have the ability to make their assessments and state the outcome over the application of both the Pre-Approval form and the Course Transfer Form. On the other hand, the department secretary will be

responsible for receiving the course request from the incoming students and notify the students the status of registering them to their courses. The requested courses by the incoming students will first be received by the coordinator. The coordinator will reject the courses if the courses are too easy for the student and only the approved courses will be sent to the department secretary. The department secretary will receive the list of the accepted courses from the coordinator. Moreover, International Student Office is responsible for receiving the language proficiency exam results from outgoing students and uploading the excel sheet of students who applied for Erasmus/Exchange and their placement information. Followingly, all related users can message each other and add the deadlines of upload and sign dates of documents as reminders to other users on their to do lists which would also be sent as emails too. Additionally, outgoing students may not register to the courses stated in their Pre-Approval Form and in such case the student must re-submit their Pre-Approval Form again to the coordinator with the registered courses. The Faculty Committee Board will get to decide if the new course will be counted toward credit during the Course Transfer Form.

Only the students who are accepted to Erasmus/Exchange can use this platform. Every accepted outgoing student will receive an email from Admin containing the link to access Bilasmus for the first time. Once the student opens the website, then the student can designate a password and login with that password for future access. Furthermore, admins can update and delete users and send transcripts of returning outgoing students to the coordinator as well. Moreover, Erasmus/Exchange coordinator can add a new course or remove a course if the coordinator decides so and does it through the application.

2.2. Functional Requirements

Authentication:

All users must be able to login to their account by using their email address and password. In case they forget their password, they must be able to reset their password via the login page, in which case the system must automatically send a password reset link to the user.

File Upload:

System must allow users to upload files to the cloud system. The system must allow coordinators to upload course transfer forms, admin to upload transcripts, and outgoing students to upload pre-approval forms, language proficiency result exams, and learning agreement documents.

File Deletion:

System will allow users to delete the files the users have uploaded themselves. System must allow coordinators to delete their uploaded course transfer forms, admins to delete uploaded transcripts, and outgoing students to delete their uploaded pre-approval forms, language proficiency result exams, and learning agreements.

Viewing Application Status:

Incoming and outgoing students must be able to view their application status, which must provide information about whether their documents are approved or not. Approval actions taken by the staff must update the application status of students.

Managing Profiles:

Profile information of all users must be manageable by admin. In addition to updating communication email addresses of users, admin must be able to delete users. Each user must have an option to view their own saved profile information, and

staff users must be able to add their signature to their own profile. Profile information of all students must be viewable by staff.

Messaging:

A messaging service must allow all users to send in-app messages to one another. Each user must be able to view incoming messages, reply to messages, search all users by name, and select a user to send a message. If a message is not seen by the receiver for a long time, the system will send an automated email to the message receiver to notify and remind about the message. Users must be able to add a message to their to-do list.

To-do Tracking:

A to-do panel must be accessible by all users for everyone to save and view their to-do tasks. To-do tasks must be added either manually by user or assigned depending on application status of students, such as whether someone sent a pre-approval form for approval or not. To-do items must have option to add remainders, in which case the system sends automated reminders at pre-determined times. Users must be able to set to-do items as completed or delete a chosen to-do item.

Notifications:

The system must send automated notifications about events and actions, such as reminders or approval of documents. Users must be able to receive and view notifications. Notifications must be delivered to the user as email as well.

Course Requests:

Incoming and outgoing students must be able to send course requests. Department secretary must have access to an interface to register courses of incoming students, and instructors must be able to view and approve requests of outgoing students. Approval status of requested courses must be saved.

Cancelling and Replacement:

After an outgoing student decides to cancel their application, the system must assign a new outgoing student from the waitlist. The system must notify the admin for them to register the user to the system manually.

Pre-approval Form:

Outgoing students must be able to send a pre-approval form. The coordinator and faculty committee board must access an interface to approve or reject the document. If an outgoing student fails to take a pre-planned course during the program, a new pre-approval form must be sent by student. The coordinator must be able to view this form and create a course transfer form accordingly, which must be approved by the faculty committee board later.

User Registration:

Admin must have access to user management to be able to register new users to the system. After a new user account is registered, a welcome email must be sent to the email address of user by the system, which must include a link to set a password for the first time.

Signing and approving:

Bilkent staff users must be able to approve or reject documents waiting for approval. Their signature saved in their profile page must be used when they sign a document digitally. Approval status of documents must be saved for viewing later.

Help:

A help page must be accessible by all users. Frequently asked questions and general information about how to use the system must be found on the help page.

2.3. Non-functional Requirements

Performance:

- The website should be loaded in less than 3 seconds.
- After each interaction on the website, any output or result of interaction should be shown in less than 3 seconds.
- The user should be logged-in in less than 5 seconds.
- The minimum time to collect and display the data needed for a page from database should be less than 1 second.
- The system should change status of a student's document status after each approval in less than 20 seconds.
- The user information, name and email edited by admin, should be updated in less than 3 seconds.
- All of the data in the system should get backed up every 1 hour.
- The in-app messages should be sent and received in less than 3 seconds.

Usability:

- The system should be user-friendly and basic for common use.
- The buttons, pop-ups, and notifications should be useful and aligned.
- Buttons and toolbar text fields should be clickable.
- The colors used in the background and foreground of each page should be consistent and not eye-straining.
- The sections and fields in each page should be related to the user based on which actor is using it.
- The help button should let the user contact admins for any technical problem.

Reliability:

- The system should be reliable such that no unexpected and unwanted events happen to the documents in the system, such as vanishing.
- System should not be down for more than an hour unless there has been a scheduled maintenance.
- The data should be preserved to prevent data loss.
- Any incident should be recorded and documented.
- System should be able to handle corner cases appearing due to user activities, such as invalid inputs and requests.

- The system should be secure as to not allow any students who are not in the Erasmus/Exchange programs to use the application. The security is meant by not allowing any outsiders to use the application.
- The system should also be secure to not allow users to pick a password that is shorter than 6 characters.

Supportability:

- The UI should appear the same to every user, that is, images, icons, etc. shouldn't change as browser changes.
- The website should be reachable by WebKit and Gecko using Browsers, such as Chrome and Safari for WebKit and Mozilla Firefox for Gecko.
- The website should be reachable on devices with at least 2 GB RAM.

3. System Models

3.1. Use-Case Diagram

The actor hierarchy can be seen in the below diagram.

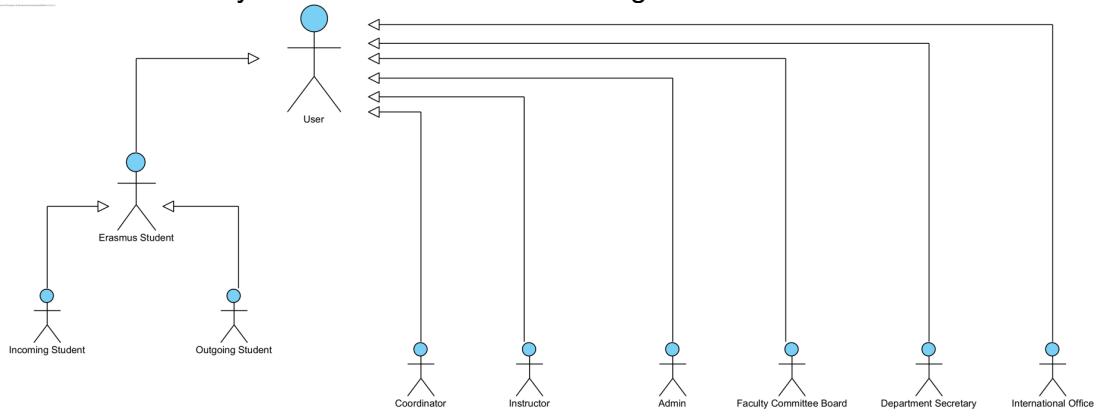


Figure 1: Actor Hierarchy

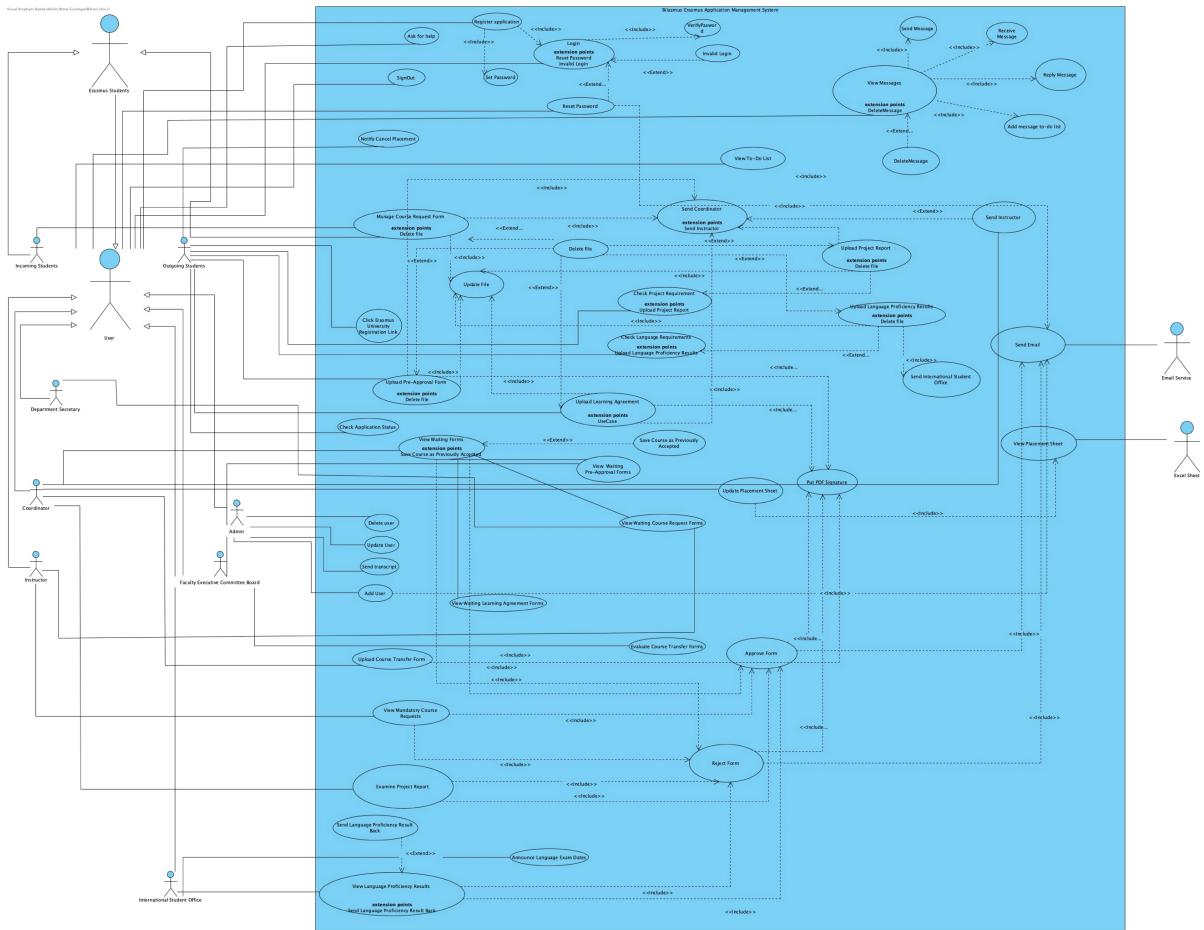


Figure 2: Use Case Diagram

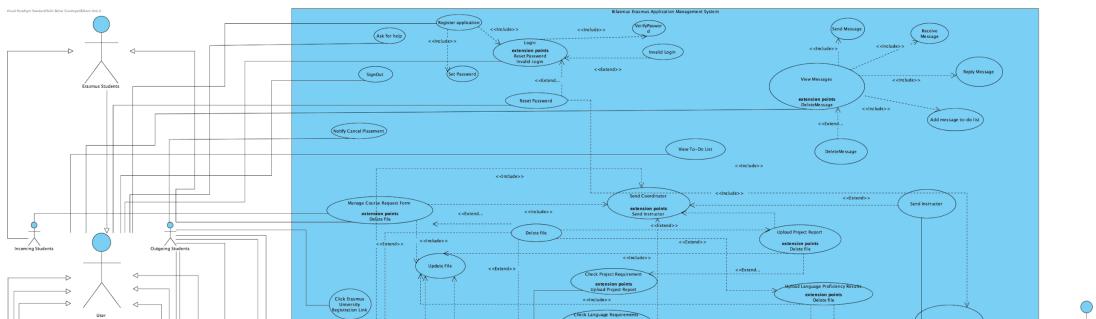


Figure 3: Close up of Use Case Diagram

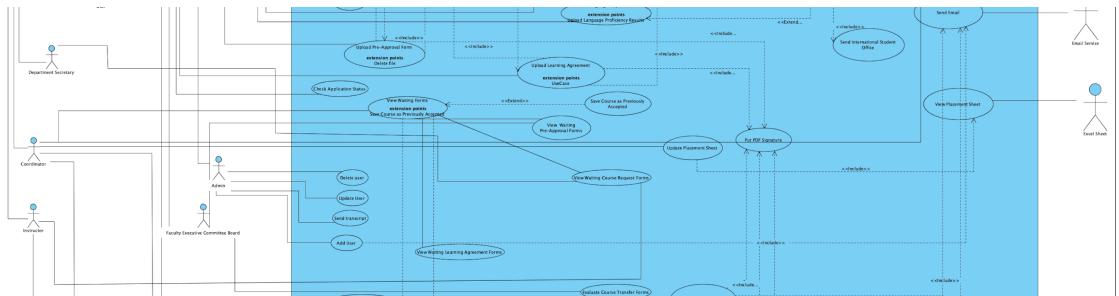


Figure 4: Close up of Use Case Diagram

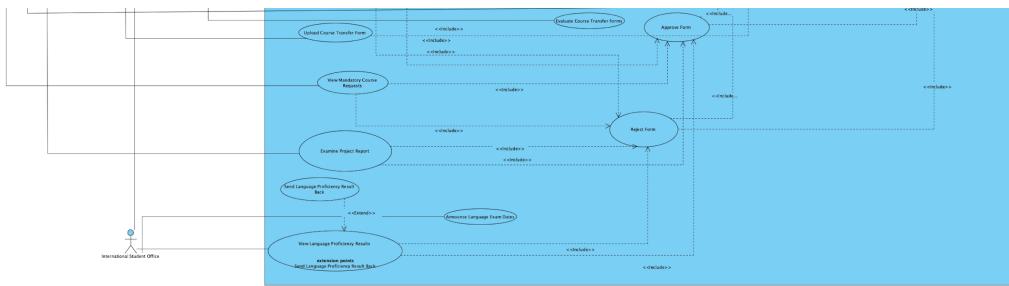


Figure5: Close up of Use Case Diagram

Use Case: Login

Participating Actor(s): User

Entry Conditions: Actor is not logged in.

Exit Conditions: Actor provides ID and password.

The Flow of Events:

- Actor clicks on ID/password fields
- Actor provides necessary information
- If information is valid actor

Use Case: Verify Password

Participating Actor(s): User, Bilasmus

Entry Conditions: User provided ID and password.

Exit Conditions: User verifies his/her password correctly.

The Flow of Events:

- User gets verification message from Bilasmus.
- User provides verification code to the Bilasmus.
- If code is valid User logs into to Bilasmus.

Use Case: Invalid Login

Participating Actor(s): User

Entry Conditions: Actor provided wrong password or ID.

Exit Conditions: This use case extends Login use case. It is initiated by the system whenever User provides wrong ID and/or password to the system.

The Flow of Events:

- Actor tries to log in.
- Actor enters invalid information to the required fields.
- The system generates an error message, and the actor must re-enter the faulty information.

Use Case: Reset Password

Participating Actor(s): User

Entry Conditions: Actor provided wrong password.

Exit Conditions: User resets his/her password successfully.

The Flow of Events:

- User gets reset password e-mail from E-mail service.
- User provides the code in the mail.
- User creates new password.
- User's password has been changed.

Use Case: Register Application

Participating Actor(s): User

Entry Conditions: Actor clicks on the registration link.

Exit Conditions: User is redirected to the website.

The Flow of Events:

- User gets reset registration e-mail from E-mail service.
- User clicks on the registration link.

Use Case: Set Password

Participating Actor(s): User

Entry Conditions: Actor is on the register page.

Exit Conditions: User sets the password.

The Flow of Events:

- User puts in the password.
- User clicks on set password button.
- User is redirected to the main page.

Use Case: Ask for Help

Participating Actor(s): User

Entry Conditions: Actor is in main page.

Exit Conditions: Actor gets helpful information and returns to the main page.

The Flow of Events:

- User clicks Help button at the top of menu bar.
- User goes help page.
- User may investigate Q&A section from help page.
- User returns to main page.

Use Case: Sign Out

Participating Actor(s): User

Entry Conditions: Actor has not signed out.

Exit Conditions: Actor signs out and leaves the program.

The Flow of Events:

- User clicks on sign out button.
- Confirmation pop-up page appears, and user selects yes or no from pop-up screen.
- If yes is selected, user signs out and leaves the program.
- If no selected, user will return to main page and stays logged in.

Use Case: View Messages

Participating Actor(s): User

Entry Conditions: Actor is in main page.

Exit Conditions: Actor can see all messages sent to him/her.

The Flow of Events:

- User encounters with message section in the main page.
- User can see all messages incoming/outgoing.

Use Case: Send Message

Participating Actor(s): User

Entry Conditions: Actor is in View All Messages section.

Exit Conditions: Actor sends message to another actor in the system.

The Flow of Events:

- Actor clicks on send message button.

- Actor selects the person who is going to receive the message.
- Actor writes his/her message in provided area.
- By clicking on “Send” button, message has gone to another actor.

Use Case: Receive Message

Participating Actor(s): User

Entry Conditions: Another actor has sent a message to main actor.

Exit Conditions: Main actor receives message from sender actor.

The Flow of Events:

- When a message received, notification will be pushed.
- Actor clicks on the person who has sent the message.
- Actor receives the message sent by another actor.

Use Case: Reply Message

Participating Actor(s): User

Entry Conditions: Actor receives a message.

Exit Conditions: Actor sends a reply message to the sender actor.

The Flow of Events:

- Actor clicks on the message bubble in the conversation.
- Actor writes his/her message as a reply to received message.
- Actor clicks on “Send” button to send reply message.
- Notification is sent to the receiver actor.

Use Case: Add Message to Todo List

Participating Actor(s): User

Entry Conditions: Actor receives a message.

Exit Conditions: A to-do list item has been added to the actor's to-do list.

The Flow of Events:

- Actor clicks on the message bubble in the conversation.
- The actor selects the message that he/she wants to add to his/her to-do list.
- The actor selects “Add to to-do List” button.
- A new to do list item has been added to the actor's to do list.

Use Case: Delete Message

Participating Actor(s): User

Entry Conditions: Actor receives a message.

Exit Conditions: Message sent to the actor has been deleted from the system.

The Flow of Events:

- Actor clicks on the message bubble in the conversation.
- Actor selects the message he/she wants to delete.
- Actor clicks on “Delete Message” button to delete message.
- That message has been deleted from the system.

Use Case: View To-Do List

Participating Actor(s): User

Entry Conditions: Actor is on the main page.

Exit Conditions: Actor reviews the to-do list.

The Flow of Events:

- Actor is on the main page.
- Actor can see all to-do items on the list.

Use Case: Notify Cancel Placement

Participating Actor(s): Outgoing Student

Entry Conditions: Actor has been placed to a university in Bilasmus system.

Exit Conditions: Actor has been removed his/her application and deleted from Bilasmus app.

The Flow of Events:

- Actor clicks on the “Cancel Placement” button.
- Confirmation pop-up which includes “Yes” and “No” buttons has been sent to the actor.
- If the actor clicks on “Yes” button, the application process will be terminated, and the actor will be deleted from the Bilasmus app.
- Information e-mail will be sent to the International Office.
- If actor clicks on “No” button, he/she will be directed to main page.

Use Case: Click Erasmus University Registration Link

Participating Actor(s): Outgoing Student

Entry Conditions: Actor has been registered to Bilasmus app.

Exit Conditions: Actor clicks on university's registration link and registers to target university

The Flow of Events:

- Actor is in the main page dedicated to himself/herself.
- Actor hits “Click University Registration Link” button.
- Actor is directed to target university's registration page.

Use Case: Manage Course Request Form

Participating Actor(s): Outgoing Student

Entry Conditions: Actor has been registered to Bilasmus.

Exit Conditions: Actor filled out the course request form.

The Flow of Events:

- Actor fills in the information about the courses he/she wants to take.
- Actor uploads the PDF version of the Syllabus of the course he/she wants to take.
- Actor chooses if the course is mandatory or elective.
- Actor hits “Submit” button and the document has been saved to Bilasmus in pdf format.

Use Case: Update File

Participating Actor(s): Outgoing Student, Incoming Students

Entry Conditions: Actor wants to update a file of the Course Request Form, Pre-Approval Form, or the Learning Agreement type and the file has not been signed by an authority figure yet.

Exit Conditions: Actor re-uploads the file successfully.

The Flow of Events:

- Actor hits “Update File” button from the system on the page of either Course Request Form, Pre-Approval Form, or the Learning Agreement Form.
- The actor uploads the file.
- Actor hits “Submit” button and the document has been saved to Bilasmus in pdf format.

Use Case: Send Coordinator

Participating Actor(s): Outgoing Student, Incoming Student

Entry Conditions: The actor has uploaded a Course Request Form, Pre-Approval Form, or a Learning agreement form. Or, the actor has uploaded a project report.

Exit Conditions: The uploaded form is successfully sent to the coordinator.

The Flow of Events:

- The actor has clicked on “Submit” button for any file of the type Course Request Form, Pre-Approval Form, or a Learning agreement.
- The message that the form is successfully sent to the coordinator is shown.

Use Case: Send Instructor

Participating Actor(s): Coordinator

Entry Conditions: The actor sends the document for the instructor to sign.

Exit Conditions: The uploaded form is successfully sent to the instructor.

The Flow of Events:

- The actor has clicked on “Send Instructor”.
- The message that the form is successfully sent to the instructor is shown.

Use Case: Delete File

Participating Actor(s): Outgoing Student, Incoming Student

Entry Conditions: Actor has uploaded a form of the kind Course Request Form, Pre-Approval Form, or Learning agreement, and the file has not been signed by an authority figure yet.

Exit Conditions: Actor's uploaded documents list will be updated and the dedicated document will be deleted from the system.

The Flow of Events:

- Actor hits “Delete” button.
- Confirmation pop-up screen, which includes “Yes” and “No” buttons will appear.
- If “Yes” is selected, the course request will be deleted, and actor's waiting approval course list will be updated.
- If “No” is selected, actor will be directed back to the course approval section.

Use Case: Check Project Requirement

Participating Actor(s): Outgoing Student

Entry Conditions: Actor has returned from Erasmus to his/her main university.

Exit Conditions: If actor has done a project in the Erasmus program, actor will be directed to “Upload Project Report” page.

The Flow of Events:

- Actor hits “Identify Project Done Status” button.
- Information screen, which includes “Yes” and “No” buttons will appear.
- If “Yes” is selected, actor will be directed to “Upload Project Report” page.
- If “No” is selected, actor will be directed back to the main page.

Use Case: Upload Project Report

Participating Actor(s): Outgoing Student

Entry Conditions: Actor has identified that he/she has done project in Erasmus program.

Exit Conditions: Actor's project report will be uploaded in pdf format successfully.

The Flow of Events:

- Actor hits “Select File” button from the system.
- Actor selects related project report from the computer.
- Actor previews the document.
- Actor hits “Submit” button and the document has been saved to Bilasmus in pdf format.

Use Case: Check Language Requirements

Participating Actor(s): Outgoing Student

Entry Conditions: Actor has been registered to Bilasmus app.

Exit Conditions: Actor's will be directed to "Upload Language Proficiency Results" page if he/she is placed a university whose education language(s) is/are not English.

The Flow of Events:

- Actor checks language requirements of the university which he/she has been placed.
- If university's education language(s) is/are different from English, the actor will be directed to "Upload Language Proficiency Result" section.

Use Case: Upload Language Proficiency Result

Participating Actor(s): Outgoing Student

Entry Conditions: Actor has been placed to a university whose education language(s) is/are different from English.

Exit Conditions: Actor's language proficiency results will be uploaded in pdf format successfully.

The Flow of Events:

- Actor hits "Select File" button from the system.
- Actor selects related language proficiency result from the computer.
- Actor previews the document.
- Actor hits "Submit" button and the document has been saved to Bilasmus in pdf format.

Use Case: Send International Student Office

Participating Actor(s): Outgoing Student

Entry Conditions: Actor has uploaded the language proficiency results.

Exit Conditions: Actor's language proficiency is successfully sent to the International Student Office.

The Flow of Events:

- Actor hits "Submit" button and the document has been saved to Bilasmus in pdf format.
- A notification saying the file has been sent successfully to the International Student Office pops up on the screen.

Use Case: Upload Pre-Approval Form

Participating Actor(s): Outgoing Student

Entry Conditions: Actor has been registered to Bilasmus app.

Exit Conditions: Actor has uploaded the Pre-Approval form successfully.

The Flow of Events:

- Actor uploads the Pre-Approval Form on the website.
- Actor hits "Submit" button.
- Since this form is an official document, the actor will be directed to "PDF Signing" section.

Use Case: Upload Learning Agreement Form

Participating Actor(s): Outgoing Student

Entry Conditions: Actor has been registered to Bilasmus app.

Exit Conditions: Actor has uploaded the Learning Agreement form successfully.

The Flow of Events:

- Actor uploads the Learning Agreement Form on the website.
- Actor hits "Submit" button.
- Since this form is an official document, the actor will be directed to "PDF Signing" section.

Use Case: Put PDF Signature

Participating Actor(s): Outgoing Student, Coordinator, Faculty Executive Committee Board

Entry Conditions: Actor either uploaded an official document or the actor must approve the official document by signing it.

Exit Conditions: Actor has successfully signed the document.

The Flow of Events:

- Actor clicks on “Add Signature” button to add signature.
- Message saying it has been successfully added pops up.

Use Case: View Waiting Forms

Participating Actor(s): Coordinator

Entry Conditions: Actor have been registered to the system and other users sent the actor files.

Exit Conditions: Actor has seen the awaiting forms.

The Flow of Events:

- Actors is on main page and sees a list of all waiting forms.

Use Case: View Waiting Pre-Approval Forms

Participating Actor(s): Coordinator, Faculty Executive Committee Board

Entry Conditions: Actor have been registered to the system and other users sent the actor Pre-Approval Forms.

Exit Conditions: Actor has seen the awaiting forms.

The Flow of Events:

- Actors sees a list of all Pre-Approval Forms.

Use Case: View Waiting Course Request Forms

Participating Actor(s): Coordinator, Instructor

Entry Conditions: Actors have been registered to the system and other users sent the actor Course Request Forms.

Exit Conditions: Actors have seen the awaiting forms.

The Flow of Events:

- Actors sees a list of all Course Request Forms.

Use Case: View Waiting Learning Agreement Forms

Participating Actor(s): Coordinator

Entry Conditions: Actor has been registered to the system and other users sent the actor Learning Agreement Forms.

Exit Conditions: Actor has seen the awaiting forms.

The Flow of Events:

- Actors sees a list of all Learning Agreement Forms.

Use Case: Save Course as Previously Accepted

Participating Actor(s): Coordinator

Entry Conditions: Actor has received courses from the Course Request Form.

Exit Conditions: Actor adds the course to the accepted courses list.

The Flow of Events:

- Actor hits “Approve New Course” button.
- Actor will input the information about the new course.
- Actors will click on “Submit” button and the accepted courses list will be updated.

Use Case: View Placement Sheet

Participating Actor(s): Coordinator, Excel Sheet

Entry Conditions: Actor is registered on Bilasmus.

Exit Conditions: Actor has seen the Placement Sheet.

The Flow of Events:

- Actors opens the “Student Wait List” page.

Use Case: Upload Course Transfer Form

Participating Actor(s): Coordinator

Entry Conditions: Actor has to receive the transcript of a specific student to prepare the course transfer form.

Exit Conditions: Actor has uploaded the Course Transfer Form.

The Flow of Events:

- Actors opens the Course Transfer Form page.
- Actor uploads the PDF of the Course Transfer Form.
- Actor chooses which outgoing student to send it to.
- Actor clicks on “Send PDF”.

Use Case: Send Email

Participating Actor(s): Email Service

Entry Conditions: A form has been rejected, approved, a registration link has been sent, or password is being reset.

Exit Conditions: Actor has successfully sent the email.

The Flow of Events:

- Actors sends an automatic email to the corresponding users to notify the user of a state.

Use Case: Update Placement Sheet

Participating Actor(s): Coordinator

Entry Conditions: Actor has been notified that an outgoing user cancelled placement at Erasmus.

Exit Conditions: Actor updates the list of placement sheet.

The Flow of Events:

- Actor opens the “Student Wait List” page.
- Actor updates the excel sheet manually on the webpage.

Use Case: Reject Form

Participating Actor(s): Coordinator, Instructor, Faculty Committee Board, International Student Office

Entry Conditions: Actors have received the file of type Course Request Form, Pre-Approval Form, Learning Agreement Form, Language Proficiency Exam Result, Course Transfer Form, Project Report.

Exit Conditions: Actors reject the course.

The Flow of Events:

- Courses waiting for approval will be listed to the actors.
- Actors hit “Reject” button.

Use Case: Approve Form

Participating Actor(s): Coordinator, Faculty Committee Board, Instructor, International Student Office

Entry Conditions: Actors have received the file of type Course Request Form, Pre-Approval Form, Learning Agreement Form, Language Proficiency Exam Result, Course Transfer Form, Project Report.

Exit Conditions: Course transfer form will be approved.

The Flow of Events:

- Courses waiting for approval will be listed to the actors.

- Actors hit “Approve” button.

Use Case: Save Course Approval Info

Participating Actor(s): Department Secretary, Coordinator, Instructor

Entry Conditions: Actors have already rejected or approved the course.

Exit Conditions: Actors’ behavior will be saved to Bilasmus.

The Flow of Events:

- Actors hit “Save” button.
- The behavior has been saved to Bilasmus.

Use Case: Add User

Participating Actor(s): Admin

Entry Conditions: Dedicated user have not been registered to the system.

Exit Conditions: Dedicated user will be registered to the system.

The Flow of Events:

- Actor hits “Add User” button.
- The actor identifies whether the user is a student or a staff member.
- Actor fills identity information regarding the user.
- Actor hits “Save” button
- Dedicated user has been registered to the system.
- Registration e-mail will be sent to dedicated user.

Use Case: Delete User

Participating Actor(s): Admin

Entry Conditions: Dedicated user has already been registered to the system.

Exit Conditions: Dedicated user has been deleted from the system.

The Flow of Events:

- Actor hits “Delete User” button.
- Actor identifies whether user a student or a staff member.
- Actor selects the to be deleted user from list of users.
- Actor hits “Save” button
- Dedicated user has been deleted from the system.

Use Case: Update User

Participating Actor(s): Admin

Entry Conditions: Dedicated user has already been registered to the system.

Exit Conditions: Dedicated user’s information has been updated.

The Flow of Events:

- Actor hits “Update User” button.
- Actor identifies whether user a student or a staff member.
- Actor selects the to be updated user from list of users.
- Actor fills necessary information to be updated.
- Actor hits “Save” button
- Dedicated user’s information has been updated in the system.

Use Case: Send Transcript

Participating Actor(s): Admin

Entry Conditions: The actor sends transcripts of Erasmus students who have returned from their Erasmus program to the coordinator.

Exit Conditions: The transcripts have been sent to the coordinator.

The Flow of Events:

- Actor clicks “Send Transcript” button.

- Actor uploads the Transcript PDF.
- Actor clicks on “Submit” button.

Use Case: Evaluate Course Transfer Form

Participating Actor(s): Coordinator, Faculty Committee Board

Entry Conditions: Student has submitted his/her course transfer form.

Exit Conditions: Course transfer form will be sent to approval or rejection.

The Flow of Events:

- Actors hit “Evaluate Course Transfer Form” button.
- Actors select the course transfer form.
- Course transfer form will be evaluated by previewing necessary documents (syllabus).

Use Case: Examine Project Report

Participating Actor(s): Coordinator

Entry Conditions: A student has uploaded his/her project report.

Exit Conditions: Project report will be examined by the coordinator.

The Flow of Events:

- Actor can see project reports.
- Actor can examine the project report by viewing the project report in a pdf format.
- Actor will “Approve” or “Reject” the form.

Use Case: View Language Proficiency Results

Participating Actor(s): International Office

Entry Conditions: Actor has been registered to the system.

Exit Conditions: Actor has viewed language proficiency results.

The Flow of Events:

- Actor selects “Check Language Proficiency Results” button.
- Actor can see the list of language proficiency results.
- Actor can preview language proficiency results in pdf format.

Use Case: Send Language Proficiency Result Back

Participating Actor(s): International Office

Entry Conditions: Actor has given a language proficiency examination to outgoing student.

Exit Conditions: Actor has sent out the results of the language proficiency exam to the outgoing students.

The Flow of Events:

- Actor uploads the language proficiency examination result files to Bilasmus.
- Actor sends the correlating form to the outgoing student by choosing which student to send to.
- Actor clicks on “Send” button.

3.2. Class Diagram

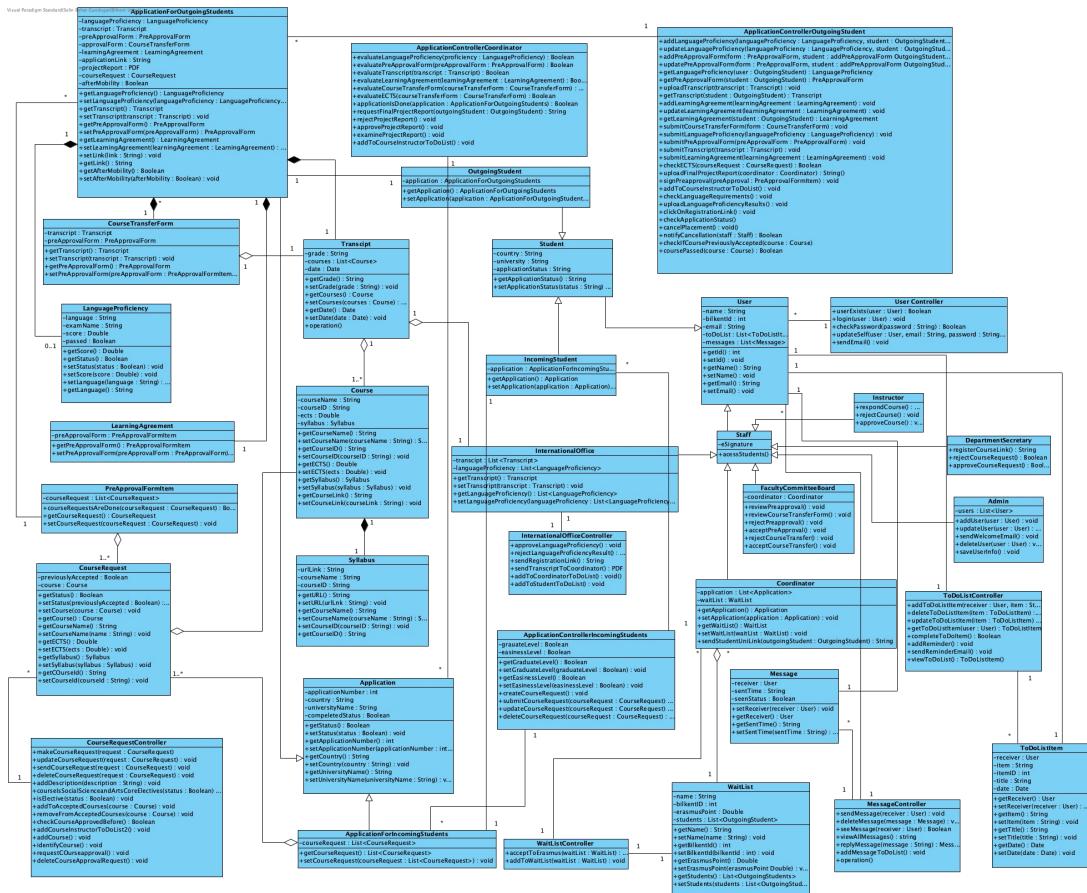


Figure 6: Bilasmus Class Diagram

User Class: User class is the parent class of Student and Staff classes. It contains the common properties of all users such as name, bilkentid, email, <List> toDoList, <List> messages of all users.

Student Class: Student class is the parent class of OutgoingStudent and IncomingStudent classes. Student class has common properties of other types of users such as the name of the Erasmus/Exchange university, the country the student is going to, and the application status.

IncomingStudent Class: This class inherits the Student Class and is a class for students coming to Bilkent University as Erasmus/Exchange students. This class has an application of the type ApplicationForIncomingStudents. This class aggregates ApplicationForIncomingStudents Class.

OutgoingStudent Class: This class inherits the Student Class and has an application for ApplicationForOutgoingStudents. Bilkent university students who will go to Exchange/Erasmus universities are of this type. This class aggregates ApplicationForOutgoingStudents Class.

Staff Class: This class inherits the User class and has the property of e-signing documents and accessing students.

UserController Class: This controller class has basic methods for all user types such as checking the user password and user email for login page and also the ability to send emails.

Instructor Class: This class extends the Staff class. Instructor class can view course request forms from outgoing students and reject or approve the courses. This class aggregates the Transcript Class.

DepartmentSecretary Class: This class is a child class of the Staff Class. It includes the properties of reviewing course request forms of incoming students to Bilkent and registering them to Bilkent courses.

Admin Class: This class is a child class of Staff Class and is the admin user of the application. It has the properties of adding, deleting, and updating users and user information. This class controls the other users in an administrative way.

Coordinator Class: This class extends the Staff Class and has the properties of viewing through the application process of all outgoing students. It also has the WaitList class to access the list and update the list accordingly.

FacultyCommitteeBoard Class: This class extends the Staff Class and has the properties of reviewing preapproval forms and the course transfer forms of all outgoing students.

InternationalOffice Class: This class extends the Staff Class and has the properties transcripts and language proficiencies of all outgoing students.

InternationalOfficeController: This is a controller class of the InternationalOffice Class, and the user has the properties of approving language proficiencies of outgoing students, and receiving the transcripts of outgoing students from their host universities after returning from their Erasmus/Exchange program. They also send the transcript to the coordinators.

Message Class: This class creates messages for users to send each other and has the properties of receiver, sent time, and seen status.

MessageController Class: This class manages the Message Class and sends, deletes, and views all messages. It also can add messages to the ToDoList of other users.

ToDoListItem Class: This class creates a ToDoListItem class, and it has the properties of receiver, item, item id, title, and date.

ToDoListController Class: This class is a control class of the ToDoListItem class and can add, and delete, update a ToDoListItem. It can also get all to do list items.

WaitList Class: This class creates a list of all students who are in the waitlist of Erasmus applications. It also has the property of Erasmus point.

WaitListController Class: This class controls the WaitList Class and either can accept students into the Erasmus program or an outgoing student back to the WaitList if the student cancels their program.

Syllabus Class: This class creates the syllabus object which has the properties a URL link, course name, and the course id.

Application Class: This class is the parent class of the other application children's classes. The class consists of common properties such as the application number, country, university name, and the completed status.

ApplicationForIncomingStudents Class: This class is a container for the course requests of incoming users, and it extends the Application Class. This class aggregates CourseRequest Class.

ApplicationControllerForIncomingStudents: This class is a controller of the ApplicationForIncomingStudents Class. This class has the properties of graduate level or easiness level status for their courses. The incoming student can create course request, update it, and delete it and send it to the Department Secretary to have them register their courses. The courses can be accepted or refused based on the status of the properties of the courses listed in the class.

ApplicationForOutgoingStudents Class: This class consists of a language proficiency exam result, transcript, pre-approval form, learning agreement, the registration link of the host university, the project report of any project courses taken in the host university, a course request, and the status of the mobility. This class has composition with LanguageProficiency, Transcript, CourseTransferForm, and LanguageAgreement classes.

ApplicationControllerForOutgoingStudents: This class is the control class of the ApplicationForOutgoingStudents Class. The class has methods to add, update, submit and delete language proficiency tests, preapproval forms, learning agreement, and course transfer form. It can also upload transcript and check ECTS of courses. It also can add things to the ToDoList, sign documents, check the application status, cancel placement, notify cancellation, add a course to the accepted courses list, and check if a course has been accepted before.

ApplicationControllerCoordinator: This class is a controller for the Coordinator Class. The class includes methods such as evaluating language proficiency, pre-approval forms, transcript, learning agreement, course transfer form, evaluate ECTS, and project reports. It can also close an application and send course requests to instructors.

Course Class: This class creates a course, and it has the properties of a Syllabus, course name, course id, and ECTS value. This class aggregates Syllabus Class.

CourseRequest Class: This class creates a course request and has the properties of previously accepted status, ECTS, course name, course id, syllabus, and course. This class aggregates the Course Class and the Courses Class.

CourseRequestController Class: This class is a controller of the CourseRequest Class. It can create, update, and delete a course request. It can also add description, identify a course, check if course is approved before, remove from accepted courses, add a description to the course, state if it is a mandatory or an elective course, and state if it is an arts or social science elective.

Transcript Class: This class consists of the GPA, the courses as a list, and the date. This class aggregates the Course Class.

LanguageProficiency Class: This class includes the properties of language, exam name, score, and the passed status of the result.

PreApprovalFormItem Class: This class contains a list of course requests. This class aggregates the CourseRequest Class.

LearningAgreement Class: This class includes the property of a PreApprovalFormItem.

CourseTransferForm Class: This class has a transcript and the pre-approval form of the outgoing student. This class aggregates the Transcript Class.

3.3. Dynamic Models Diagram

3.3.1. State Machine Diagrams

3.3.1.1. Learning Agreement Upload

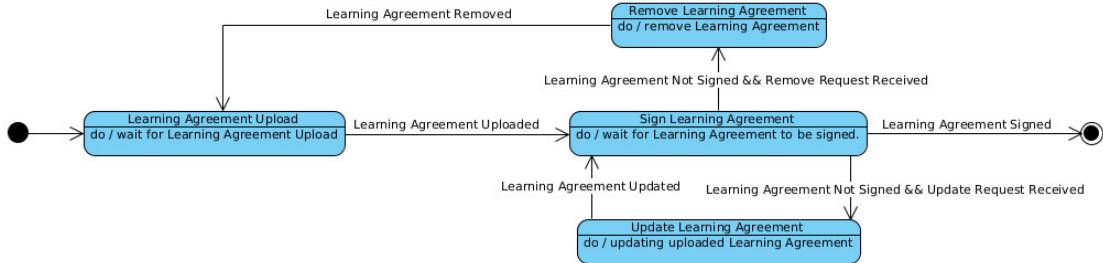


Figure 7: Learning Agreement Upload State Machine Diagram

This state machine diagram is chosen because it contains extensive, detailed, and one of the most crucial parts of the program. It models all states of the outgoing students' application process before and after mobility. It starts with registering via a link and ends with the course transfer form being uploaded.

3.3.1.2. Course Request Process

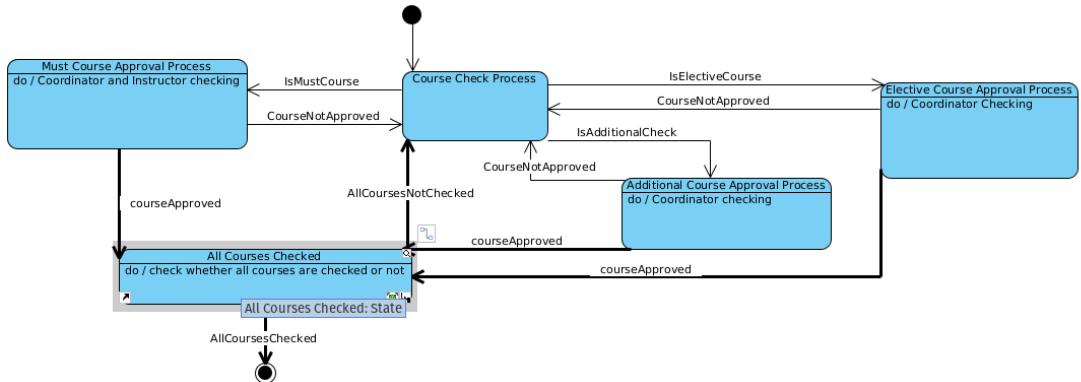


Figure 8: Course Request Process State Machine Diagram

This state machine diagram is a sub-state machine diagram of the previous state machine diagram. It models the Course Request Process state of the previous diagram, starting with identifying the course type and ending with all courses being checked.

3.3.2. Activity Diagrams

3.3.2.1. Course Transfer Form Evaluation

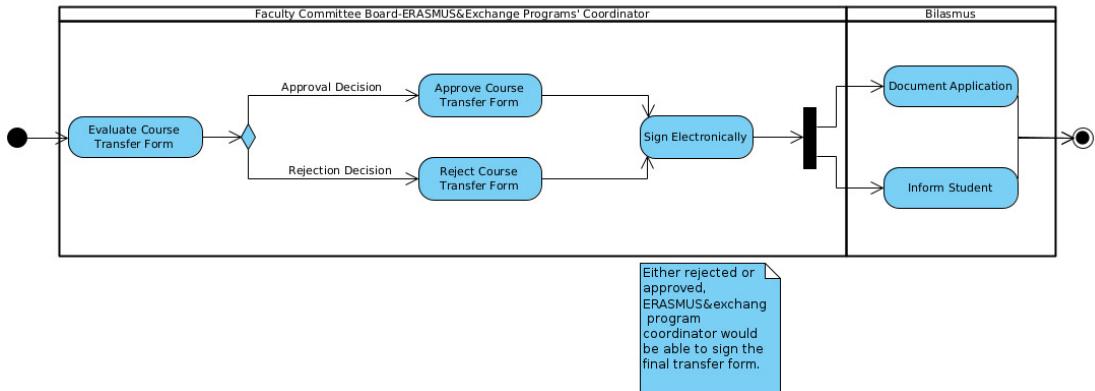


Figure 9: Course Transfer Form Evaluation Activity Diagram

This diagram represents how the course transfer form evaluation should proceed. We implemented this diagram to describe the process necessary for the application to end. It also gives an idea of what the Bilasmus will and will not do.

3.3.2.2. Course Identification

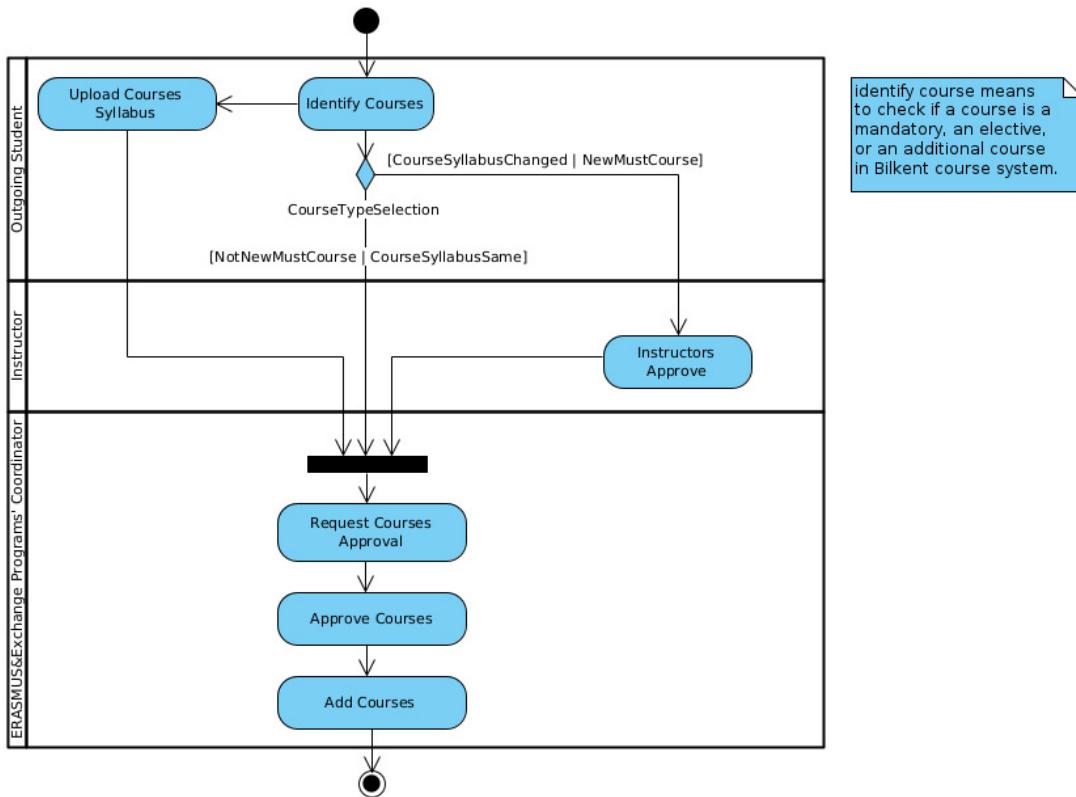


Figure 10: Course Identification Activity Diagram

This diagram represents how the courses are approved, and additionally, it shows how different courses are treated in the process. The primary motivation behind implementing this diagram was to visualize each step in the course approval process.

3.3.3. Sequence Diagrams

3.3.3.1. Course Requesting

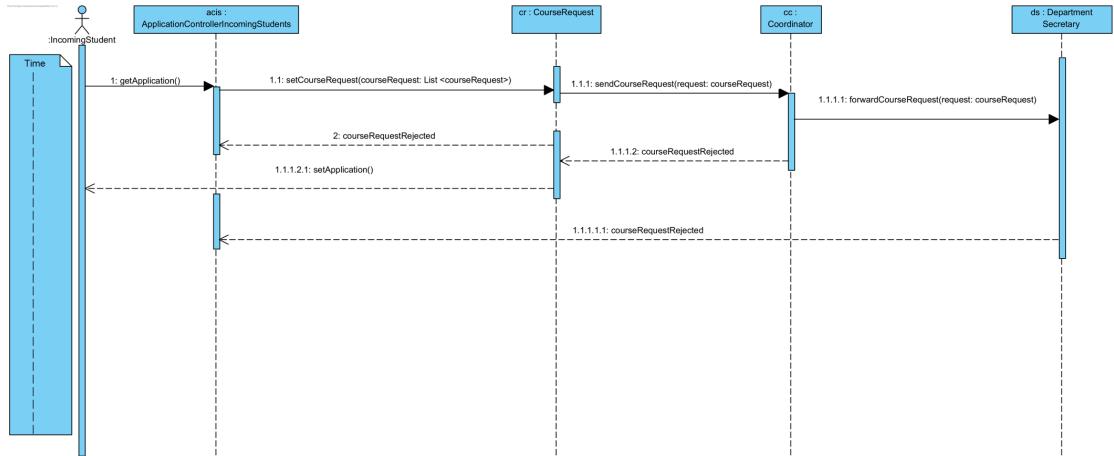


Figure 11: Course Requesting Sequence Diagram

This diagram represents requesting course approval will work in low-level design.

3.4. User Interface

The UI Flow is shown below.

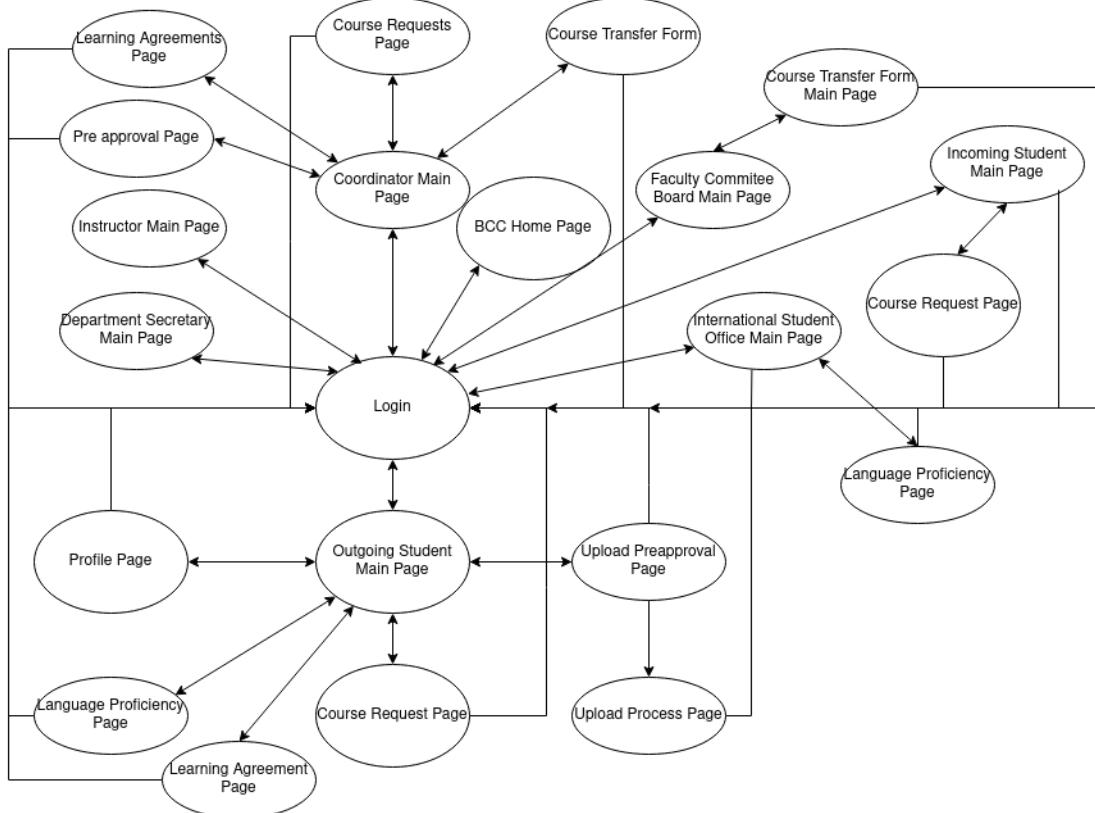


Figure 12: The UI Flow

Login Page



Figure 13: Login page

Login Failed Page

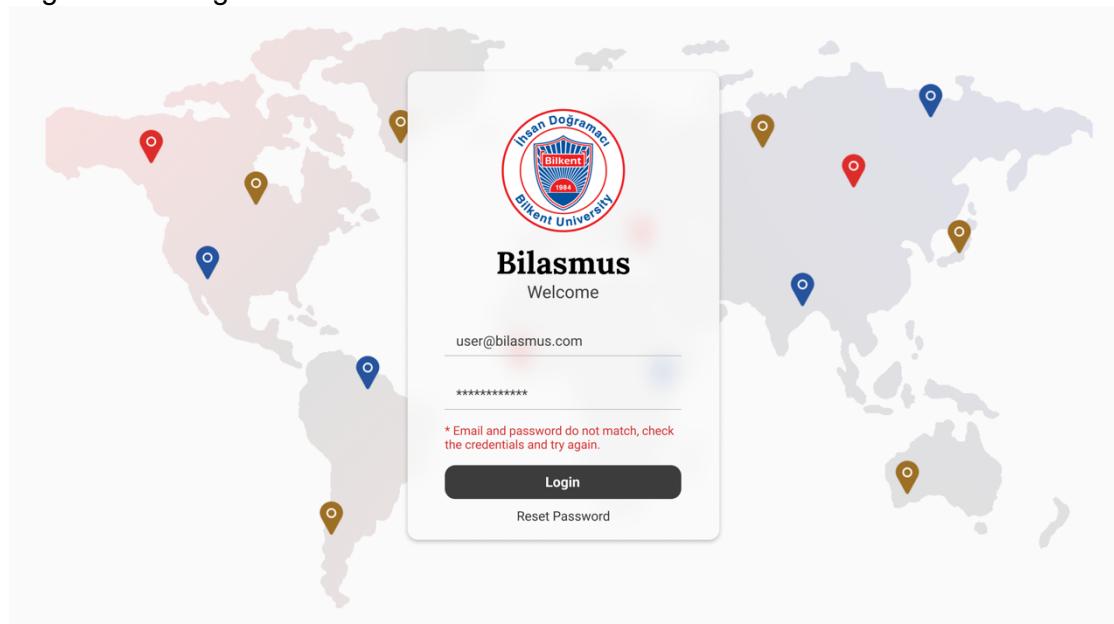


Figure 14: Login failed page

Reset Password Page

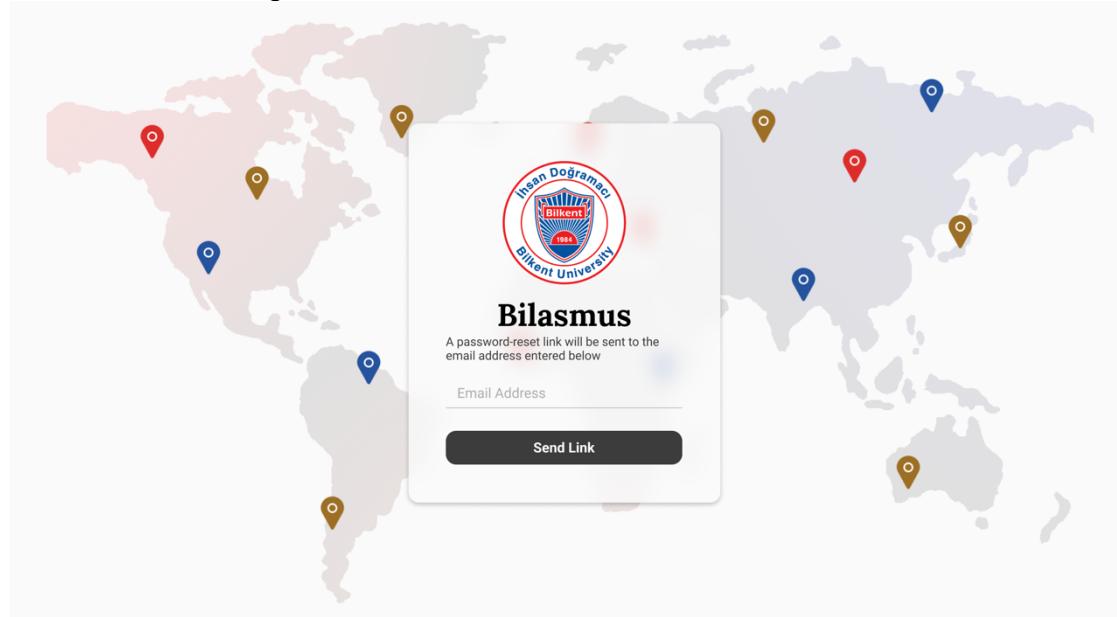


Figure 15: Login failed page

Profile Page

A screenshot of a user profile page. At the top, there is a red navigation bar with icons for Home, Course Request, Pre-Approval Form, Learning Agreement, Language Proficiency, Help, Log out, a message icon, a notification bell icon, and a user profile icon. The main content area has a light gray background. In the upper right corner, there is a circular profile picture of a woman holding a book. To the left of the picture, the word "Profile" is written in bold black text. Below the profile picture is a light gray rounded rectangle containing the user's information: Name: Student1, BilkentID: 22xxxxxx, Phone Number: +90 xxx xxx xx xx, Citizen ID: xxxxxxxxxxxx, and Email: student1@ug.bilkent.edu.tr. At the bottom of the page, there are two buttons: "Contact BCC to update info" and "Contact BCC" (in a darker shade), and two more buttons: "Add signature" and "Upload Signature".

Figure 16: Profile page template of all users

Notifications Page

The screenshot shows the Notifications Page template. At the top, there is a navigation bar with links for Home, Course Request, Preapproval Form, Learning Agreement, Language Proficiency, and Help. On the far right of the navigation bar are Log out, a message icon, a bell icon, and a user profile picture. Below the navigation bar, there is a section titled "Learning Agreement" with a placeholder for uploading a document. A large button labeled "Choose File" is available to select a file. To the right of this, a sidebar titled "Notifications" displays three items: a green checkmark indicating a pre-approval was approved, a purple reminder for re-uploading a learning agreement, and a red X indicating a learning agreement was rejected.

Learning Agreement
Upload your Learning Agreement document by either dragging and dropping your file to the box below, or clicking on the Choose File button.

Drop your file here...

Choose File

Previously Uploaded Files

	Name Surname Learning Agreement.pdf	Download
	Uploaded: 1 November 2022, 13:00	
	Status: Approved	Delete

Notifications

[NEW] 24 minutes ago
Your Pre-Approval form has been approved!

Yesterday, at 18:00
Reminder: Re-upload Learning Agreement*

28 October, at 18:00
Your Learning Agreement has been rejected!

Figure 17: Notifications page template of all users

Help

The screenshot shows the Help page template. At the top, there is a navigation bar with links for Home, Course Request, Preapproval Form, Learning Agreement, Language Proficiency, and Help. On the far right of the navigation bar are Log out, a message icon, a bell icon, and a user profile picture. Below the navigation bar, there is a "Welcome" message and an "Application Status" section. On the right side, there is a live chat interface with a text input field and a "Send" button.

Welcome

Application Status

How can we help you?

Type here...

Send

Figure 18: Help page template of all users

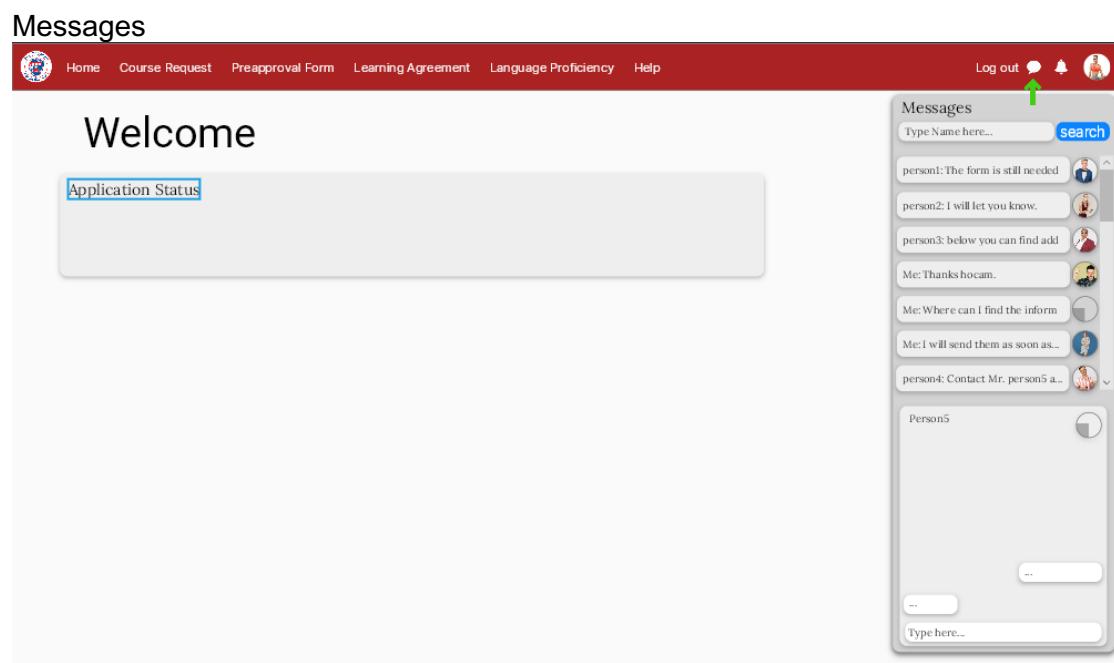


Figure 19: Message template of all users

The screenshot shows a 'Detailed Info' page for a user named 'Student Name 1'. It includes:

- Profile:** Department: Computer Engineering, Phone Number: +90 xxx xxx xx xx, Bilkent ID: 22xxxxxx, Email: name.surname@ug.bilkent.edu.tr
- Signed Documents:** Document 1, Document 2, Document 3
- All Uploaded Documents:** Document 1, Document 2, Document 3, Document 4, Document 5, Document 6, Document 7
- Course List:** Course List is Empty

Figure 20: Detailed info page of all users

Main Page- Outgoing Student

Welcome

Application Status

Completed ✓

- ✓ Upload Course List
- ✓ Upload preapproval form
- ✓ Upload learning agreement
- ✓ Upload Transcript
- ✓ Upload Course Transfer Form

General Overview

UNIVERSITY OF ROME	Digital Design	Details	Syllabus
UNIVERSITY OF ROME	Computer Architecture	Details	Syllabus
UNIVERSITY OF ROME	Introduction to Roman History	Details	Syllabus
UNIVERSITY OF ROME	Orientation	Details	Syllabus

Useful Links

Bilkent Erasmus Page
<https://www.bilkent.edu.tr...>

Erasmus+ Application System
<https://app.erasmus.bilkent.edu.tr/>

To-Do List

Your To-Do List is empty

Figure 21: Main page of outgoing student users

Course Request Page- Outgoing Student

Add a course to the course request list

Course ID	Course Name	Course Webpage Link	Course Type	Course to Satisfy	Course Syllabus
Course ID	Course Name	Course Link	Elective	CS315	<input type="button" value="Browse..."/> No file selected.

Added Course Requests

HIST150 - History of Germany ECTS: 7.5 Previously Approved: Yes

Send Course Request List

Figure 22: Course Request page of outgoing student users

Language Proficiency Page- Outgoing Student

The screenshot shows the Language Proficiency page for outgoing student users. At the top, there is a red navigation bar with icons for Home, Course Request, Preapproval Form, Learning Agreement, Language Proficiency (which is the active tab), and Help. To the right of the navigation bar are Log out, message, bell, and user profile icons. Below the navigation bar, the main content area has a title 'Welcome' and a large 'Application Status' box. In the center of the page, there is a file upload interface showing a progress bar labeled 'Loading...' and a green 'Upload' button. A small green arrow points upwards towards the upload area. To the right of the upload area is a 'To-Do List' section with two items: 'Upload preapproval form' and 'Upload learning agreement'. Below the upload area, a message box displays 'Successfully uploaded' and an 'Ok' button.

Figure 23: Language Proficiency page of outgoing student users

Pre-Approval Form Page- Outgoing Student

The screenshot shows the Pre-Approval Form page for outgoing student users. At the top, there is a red navigation bar with icons for Home, Course Request, Pre-Approval Form (which is the active tab), Learning Agreement, Language Proficiency, and Help. To the right of the navigation bar are Log out, message, bell, and user profile icons. Below the navigation bar, the main content area has a title 'Pre-Approval Form'. It includes a note: 'Upload your Pre-Approval Form document by either dragging and dropping your file to the box below, or clicking on the Choose File button.' Below this note is a large input field with a drop icon and the placeholder text 'Drop your file here...'. Below the input field is a 'Choose File' button. At the bottom of the page, there is a section titled 'Previously Uploaded Files' containing a list of one file: 'Name Surname Preapproval Form.pdf' (uploaded on 1 November 2022, 13:35, status: Approved). It includes a 'Download' button and a 'Delete' button.

Figure 24: Pre-Approval Form page of outgoing student users

Pre-Approval Form Page- Upload Process- Outgoing Student

The screenshot shows the 'Pre-Approval Form' section of the application. At the top, there is a navigation bar with links for Home, Course Request, Pre-Approval Form, Learning Agreement, Language Proficiency, and Help. On the far right of the navigation bar are icons for Log out, a message bubble, a bell, and a user profile. Below the navigation bar, the title 'Pre-Approval Form' is displayed in bold. A sub-instruction 'Upload your Pre-Approval Form document by either dragging and dropping your file to the box below, or clicking on the Choose File button.' follows. A large, light-gray rectangular area contains a file preview for 'Name Surname Preapproval Form.pdf'. The preview includes the file name, upload date ('Uploaded: 1 November 2022, 15:01'), and status ('Status: Pending Upload'). Below this preview is a small text instruction 'Click on the button below to upload'. A dark gray button labeled 'Upload' is centered at the bottom of this area. Further down, under the heading 'Previously Uploaded Files', there is a preview for another file, 'Name Surname Preapproval Form.pdf', showing its upload details ('Uploaded: 1 November 2022, 13:35') and status ('Status: Approved'). To the right of this preview are two buttons: 'Download' (in black) and 'Delete' (in red).

Figure 25: Pre-Approval Form page- file upload process- of outgoing student users

Learning Agreement Page- Outgoing Student

The screenshot shows the 'Learning Agreement' section of the application. At the top, there is a navigation bar with links for Home, Course Request, Pre-Approval Form, Learning Agreement, Language Proficiency, and Help. On the far right of the navigation bar are icons for Log out, a message bubble, a bell, and a user profile. Below the navigation bar, the title 'Learning Agreement' is displayed in bold. A sub-instruction 'Upload your Learning Agreement document by either dragging and dropping your file to the box below, or clicking on the Choose File button.' follows. A large, light-gray rectangular area contains a file upload interface. It features a central icon of a laptop with an upward arrow, above which is the text 'Drop your file here...'. Below this is a dark gray button labeled 'Choose File'. Underneath the upload area, there is a section titled 'Previously Uploaded Files'. It displays a preview for 'Name Surname Learning Agreement.pdf', showing its upload details ('Uploaded: 1 November 2022, 13:00') and status ('Status: Approved'). To the right of this preview are two buttons: 'Download' (in black) and 'Delete' (in red).

Figure 26: Learning Agreement page of outgoing student users

Main Page-Incoming Student

The screenshot shows the main page for incoming students. At the top, there is a navigation bar with links for Home, Course Request, and Help, along with a Log out button and a user profile icon. The main content area is divided into two sections: 'Course List' and 'To-Do List'. The 'Course List' section displays a grid of courses from 'İHSAN DOĞRAMACI BİLKENT ÜNİVERSİTESİ'. Each course row includes the course name, university name, 'Details' button, and 'Syllabus' button. The 'To-Do List' section is currently empty, displaying the message 'Your To-Do List is Empty'.

Figure 27: Main page of incoming student users

Course Request- Incoming Student

The screenshot shows the 'Course Request' page for incoming students. At the top, there is a navigation bar with links for Home, Course Request, and Help, along with a Log out button and a user profile icon. The main content area is titled 'Course Request' and contains a form for adding a course to the request list. The form has fields for 'Course ID' and 'Course Name', both with input boxes. Below the form is a green 'Add' button. Further down, there is a section titled 'Added courses' which lists five courses: CS319, LNGT11, FA171, CS465, and CS426. Each course entry includes an 'Update' button (orange) and a 'Delete' button (red).

Figure 28: Learning Agreement page of outgoing student users

Main Page- Coordinator

Welcome

Waiting Course Approvals

Student Profile	Course Request Details	Instructor Approval Status
Student Name 1 University of Rome	Digital Design ECTS: 4	✓ ✗ Instructor Approval: ✓
Student Name 1 University of Rome	Orientation ECTS: 2	✓ ✗ Instructor Approval: Not Required
Student Profile	Course Request Details	Instructor Approval Status
Student Name 2 Newyork University	Quantum Physics I ECTS: 3	✓ ✗ Instructor Approval: ✓
Student Name 2 Newyork University	Processor Design ECTS: 6	✓ ✗ Instructor Approval: 🔍
Student Name 2 Newyork University	Advanced Calculus II ECTS: 4	✓ ✗ Instructor Approval: ✗

To-Do List
Your To-Do List is Empty

Figure 29: Main page of coordinator users

Course Request Page- Coordinator

Coordinator

Course Requests

Course Request Details	Action Buttons
Tilburg University - Social Media - COMD 101	Check Request Approve Reject
University of Amsterdam - Algorithms I - CS 101	Check Request Approve Reject
Technical University of Berlin - Game Theory - IE 356	Check Request Approve Reject
University of Padova - Italian History - HIST 104	Check Request Approve Reject
University of Groningen - Social Behaviour - PYSC 107	Check Request Approve Reject
Politecnico di Milano - Mechanical Design - ME 281	Check Request Approve Reject

Search Course Transfer Form

Student Name Search

Figure 30: Course Request page of coordinator users

Course Request Page-View Check Request- Coordinator

Student 1		Course Type	Evaluated by Instructor	Course Link and Syllabus	Final Decision
Course Name		Mandatory	Approved	Link Syllabus	Approve Reject
1. CE319- Object-Oriented Software Engineering		Elective	-	Link Syllabus	Approve Reject
2. LNG170- German I		Elective	-	Link Syllabus	Approve Reject
3. FA71- Introduction to Art Design, and Culture I		Mandatory	-	Link Syllabus	Approve Reject
4. CE465- Computer Graphics		Mandatory	Rejected	Link Syllabus	Approve Reject
5. CE426- Parallel Computing		Mandatory	-	Link Syllabus	Approve Reject

Student 2		Course Type	Evaluated by Instructor	Course Link and Syllabus	Final Decision
Course Name		Mandatory	Approved	Link Syllabus	Approve Reject
1. ART203- Renaissance Art		Elective	-	Link Syllabus	Approve Reject
2. LANG120- German II		Elective	-	Link Syllabus	Approve Reject
3. MAT360- Linear Algebra		Mandatory	Rejected	Link Syllabus	Approve Reject
4. IEI01- Introduction to Industrial Engineering		Mandatory	-	Link Syllabus	Approve Reject
5. PSYC320- Psychology of Consumerism		Mandatory	-	Link Syllabus	Approve Reject

Student 3		Course Type	Evaluated by Instructor	Course Link and Syllabus	Final Decision
Course Name		Mandatory	Pending	Link Syllabus	Approve Reject
1. HIST150- History of Germany		Elective	-	Link Syllabus	Approve Reject
2. LNG140- Urdu I		Elective	-	Link Syllabus	Approve Reject
3. FA101- Introduction to Fashion		Mandatory	Rejected	Link Syllabus	Approve Reject
4. MAN220- Business Law		Mandatory	-	Link Syllabus	Approve Reject
5. ECON410- Microeconomic Theory		Mandatory	-	Link Syllabus	Approve Reject

Figure 31: Course Request page of coordinator users

Pre-Approval Page- Coordinator

 Name Surname Preapproval Form.pdf	Student: Name Surname	Uploaded: 1 November 2022, 13:35	View	Approve	Reject
 Name Surname Preapproval Form.pdf	Student: Name Surname	Uploaded: 2 November 2022, 14:25	View	Approve	Reject
 Name Surname Preapproval Form.pdf	Student: Name Surname	Uploaded: 3 November 2022, 15:15	View	Approve	Reject
 Name Surname Preapproval Form.pdf	Student: Name Surname	Uploaded: 4 November 2022, 16:05	View	Approve	Reject
 Name Surname Preapproval Form.pdf	Student: Name Surname	Uploaded: 5 November 2022, 17:55	View	Approve	Reject

Figure 32: Pre-Approval Form page of coordinator users

Learning Agreement Page- Coordinator

The screenshot shows a list of five learning agreements for review:

- Name Surname Learning Agreement.pdf
Student: Name Surname
Uploaded: 1 November 2022, 13:35
View Approve Reject
- Name Surname Learning Agreement.pdf
Student: Name Surname
Uploaded: 2 November 2022, 14:25
View Approve Reject
- Name Surname Learning Agreement.pdf
Student: Name Surname
Uploaded: 3 November 2022, 15:15
View Approve Reject
- Name Surname Learning Agreement.pdf
Student: Name Surname
Uploaded: 4 November 2022, 16:05
View Approve Reject
- Name Surname Learning Agreement.pdf
Student: Name Surname
Uploaded: 5 November 2022, 17:55
View Approve Reject

Figure 33: Learning Agreement page of coordinator users

Learning Agreement Page-No Requests- Coordinator

The screenshot shows a message indicating there are no pending approvals:

No Pending Approval
There is not any Learning Agreement that needs your attention. New approval tasks will appear here once they are assigned to you.

Figure 34: Learning Agreement page-no requests- of coordinator users

Course Transfer Forms Page-Coordinator

The screenshot shows a list of 11 students (Person1 to Person11) with their respective email addresses and profile pictures. Each student entry includes a 'Files...' button, a 'Loading...' status indicator, an 'Upload' button, and a download icon. At the top right of the list area, there is a green arrow pointing upwards towards a 'Download Forms' button.

Figure 35: Course Transfer Forms page of coordinator users

Main Page- International Student Office

The screenshot displays the main interface for International Student Office users. It includes a 'Send Transcript' section with a search bar and a 'Send Transcript to Coordinator' button. Below this are two 'Add Todo List' sections: one for 'Coordinator's todo List' and one for 'Student's todo List', both with fields for 'Title', 'Due Date', and 'Add Reminder' options. On the right side, there is a 'To-Do List' sidebar with checkboxes for 'Check Language Proficiency Results' and 'Send Registration Link'. At the bottom, there is a 'Registration Link' section with a 'Student ID' input field and a 'Send Registration Link' button.

Figure 36: Main page of international student office users

Language Proficiency Page- International Student Office

The screenshot shows the Language Proficiency Page for International Student Office users. At the top, there are links for Home, Evaluate Language Proficiency, and Help, along with a Log out button and user icons.

Student Entries:

- Student1: exam name: IELTS, language: English, score: 7.5. Passed: Yes No
- Student2: exam name: TOEFL IBT, language: English, score: 115. Passed: Yes No
- Student3: exam name: TEF, language: French, score: 72. Passed: Yes No
- Student4: exam name: DSD, language: German, score: CI. Passed: Yes No
- Student5: exam name: IELTS, language: English, score: 6.5. Passed: Yes No

Approved Students:

- Student1
- Student2
- Student3
- Student4
- Student5
- Student6
- Student7
- Student8
- Student9
- Student10
- Student11
- Student12
- Student13

Rejected Students:

- Student1
- Student2
- Student3
- Student4
- Student5
- Student6
- Student7
- Student8
- Student9
- Student10
- Student11
- Student12
- Student13

Figure 37: Language Proficiency page of international student office users

Main Page- Department Secretary

The screenshot shows the Main Page for Department Secretary users. At the top, there are links for Home, Course Request, and Help, along with a Log out button and user icons.

Welcome

Course Requests

Student Name	Course Name	Status
Student Name 1	Course Name 1	✓ ✗
Student Name 1	Course Name 2	✓ ✗
Student Name 2	Course Name 3	✓ ✗
Student Name 2	Course Name 4	✓ ✗
Student Name 2	Course Name 5	✓ ✗

To-Do List

Your To-Do List is Empty

Figure 38: Main page of department secretary users

Course Requests Page- Department Secretary

Course Requests

Student 1

- CS319- Object-Oriented Software Engineering
- LNC171- Turkish I
- FA171- Introduction to Art Design, and Culture I
- CS465- Computer Graphics
- CS426- Parallel Computing

Student 2

- CS315- Programming Languages
- CS342- Operating Systems
- FA171- Introduction to Art Design, and Culture I
- CS475- Data Privacy
- CS476- Automata Theory and Formal Languages

Student 3

- SOC101- Introduction to Sociology
- LNC172- Turkish II
- PSYC100- Introduction to Psychology
- ECON101- Introduction to Economics
- COMD341- Media and Society

Approve Reject

Approve Reject

Approve Reject

Approve Reject

Approve Reject

Approve Reject

Figure 39: Course Request page of department secretary users

Main Page-Faculty Committee Board

Welcome

Preapproval List

UNIVERSITY NAME 1 Student Name 1	View
UNIVERSITY NAME 2 Student Name 2	View
UNIVERSITY NAME 3 Student Name 3	View
UNIVERSITY NAME 4 Student Name 4	View
UNIVERSITY NAME 5 Student Name 5	View

Course Transfer Forms

UNIVERSITY NAME 1 Student Name 1	View
UNIVERSITY NAME 2 Student Name 2	View
UNIVERSITY NAME 3 Student Name 3	View
UNIVERSITY NAME 4 Student Name 4	View
UNIVERSITY NAME 5 Student Name 5	View
UNIVERSITY NAME 6 Student Name 6	View

To-Do List

Your To-Do List is Empty

List of Signatures

List of Signatures

Figure 40: Main page of faculty committee board users

Course Transfer Form Page- Faculty Committee Board

The screenshot shows a web application interface for managing course transfer forms. At the top, there is a navigation bar with links for 'Home', 'Help', and 'Log Out'. On the right side of the header are icons for messaging, notifications, and user profile. Below the header, the main content area is titled 'Faculty Committee Board'.

The central feature is a section titled 'Course Transfer Form' containing six rows, each representing a student's form. Each row has a 'Student' input field (containing 'Student 1' through 'Student 6'), a 'Check Form' button, an 'Approve' button (green), and a 'Reject' button (red). To the right of this main section is a sidebar titled 'Search Course Transfer Form' with a search input field and a 'Search' button.

Figure 41: Course Transfer Form page of faculty committee board users

Pre-Approval Forms Page- Faculty Committee Board

The screenshot shows a web application interface for managing pre-approval forms. At the top, there is a navigation bar with links for 'Home', 'Help', and 'Log out'. On the right side of the header are icons for messaging, notifications, and user profile.

The main content area is titled 'Pre-Approval Forms' and displays five entries, each representing a pre-approval form. Each entry includes a document icon, the file name 'Name Surname Preapproval Form.pdf', the student's name 'Surname', and the upload date ('Uploaded: 1 November 2022, 13:35'). To the right of each entry are three buttons: 'View' (grey), 'Approve' (green), and 'Reject' (red).

Figure 42: Pre-Approval Forms page of faculty committee board users

Main Page- Instructor

The screenshot shows the main page for instructor users. At the top, there is a header with a logo, 'Home', 'Help', 'Log out', and a user icon. Below the header, the title 'Course Requests' is displayed. The page is divided into two sections: 'Student 1' and 'Student 2'. Each section contains a table with course details, course type, and action buttons for 'Link', 'Syllabus', 'Approve', and 'Reject'. In 'Student 1', three courses are listed: CE319- Object-Oriented Software Engineering, CE465- Computer Graphics, and CE426- Parallel Computing, all marked as Mandatory. In 'Student 2', one course, CS359-Operating Systems, is listed as Mandatory.

Figure 43: Main page of instructor users

Main Page- Admin

The screenshot shows the main page for admin users. At the top, there is a header with a logo, 'Home', 'Transcript', 'Help', 'Log Out', and a user icon. The main area is titled 'Welcome Admin' and contains three main sections: 'Add User', 'Delete User', and 'Update User'. The 'Add User' section has fields for Name, Bilkent ID, and E-mail, with radio buttons for 'Staff' and 'Student' roles, and a 'Add User' button. The 'Delete User' section has a field for Bilkent ID and a red 'Delete User' button. The 'Update User' section has fields for Bilkent ID, E-mail, and New E-mail, with a 'Update User' button. To the right, there is a 'To-Do List' with checkboxes for 'Add new incoming students' and 'Delete students who cancelled'.

Figure 44: Main page of Admin

Transcript Page- Admin

The screenshot shows a web-based application for managing transcripts. At the top, there's a navigation bar with links for Home, Transcript, and Help, along with Log Out and user profile icons. Below the navigation is a search bar with a placeholder 'Type here...' and a 'search' button. To the left of the search bar is a vertical list of categories from A to Z, with 'A' currently selected. Under category A, there are eleven entries, each consisting of a small profile icon, the name 'Person1' through 'Person11', and the corresponding email address 'Person1@gmail.com' through 'Person11@gmail.com'. To the right of the main list, there's a separate box containing a profile icon for 'Person1', a folder icon labeled 'Files...', and a green 'Upload' button.

Figure 45: Transcript page of Admin

3.5. Improvement Summary

In conclusion, the Analysis Report was corrected based on the given feedback. The class, sequence, and state diagrams were corrected along with a few grammatical errors in the first iteration. The use case was modified based on the feedbacks and its verbal explanation was also added. The user interface mock-ups were also changed as new features were added and style of some functionalities were improved. Furthermore, no new additional requirements of functional or non-functional types were added to the report. Thus, those sub-categories were not modified.