

BILKENT UNIVERSITY
ENGINEERING FACULTY
DEPARTMENT OF COMPUTER ENGINEERING

CS 319

ANALYSIS REPORT

Emirkan Derköken	22002693
Murat Güney Kemal	22002692
Cengizhan Terzioğlu	22003014
Göktuğ Kuşçu	22002867
Bora Yılmaz	22003359
Onur Asım	21903375

Instructor: Eray Tüzün

Table of Contents

1. INTRODUCTION	1
2. PROPOSED SYSTEMS	2
2.1 Functional Requirements.....	2
2.1.1 Authentication & Authorization	2
2.1.2 User Types.....	2
2.1.2.1 Admin	2
2.1.2.2 International Office	3
2.1.2.3 Student.....	3
2.1.2.4 Erasmus Coordinator.....	4
2.1.2.5 Course Coordinator.....	5
2.1.2.6 Chair	5
2.1.2.7 Dean/Director.....	5
2.1.3 Application Forms.....	6
2.1.4 Features	6
2.1.4.1 To Do List.....	6
2.1.4.2 Calendar	7
2.1.4.3 Messenger.....	7
2.1.4.4 Application Assistant.....	7
2.2 Non-Functional Requirements	8
2.2.1 Usability.....	8
2.2.2 Reliability	8
2.2.3 Performance.....	8
2.2.4 Supportability	8
2.2.5 Security.....	8
2.3 Pseudo Requirements.....	9
2.4 System Models	10
2.4.1 Use Case Model.....	10
2.4.2 Dynamic Models	16
2.4.2.1 State Diagrams.....	16
2.4.2.2 Activity State Diagram.....	18
2.4.2.3 Sequence Diagram.....	19
2.4.3 Object and Class Models.....	20
2.4.4 User Interface	22

1. INTRODUCTION

In this project, Bilkent University's Erasmus web page is going to be redesigned according to the requests of the Erasmus coordinators. This web application aims to make the whole application process easier for both the students and the coordinators. We are also digitalizing the Erasmus application forms in order to reduce the number of papers and working hours wasted.

In this application, students are going to be able to make and follow their applications step by step, see important deadlines and ask their questions to the coordinators through direct messages. Students are also going to be able to choose their Erasmus courses -from an approved list of courses- and they will have the ability to propose new course matchings to the course coordinators through this application. Our application assistant will guide students throughout all these steps and make recommendations & predictions based on past data. Finally, coordinators are going to be able to see applications, approve courses, message students and use their e-signature to approve documents.

2. PROPOSED SYSTEMS

2.1 Functional Requirements

2.1.1 Authentication & Authorization

Authentication and authorization functionalities will be covered up by the framework that is going to be used to build this application. Authentication information of every user will be registered by the admin users, whose authentication information are embedded into the application and have access to the admin panel, according to the clients' request and will be stored in the application's designated database. The authentication functionality will be handled during the initial state of the application which is the login phase, thus there are possibly going to be separate login pages for different user types. Authorization functionality of this application will be governed by user types, which means that every user will be authorized according to his/her user type and that type's pre-determined authorization.

2.1.2 User Types

User is the common object that will bring in authentication and authorization functions to the project. Users will be able to have one or more roles that will allow them to have different authorizations. Role objects will hold a role code to determine the specific role. Role codes include 1) admin, 2) international office, 3) student, 4) coordinator, 5) instructor, 6) chair and 7) dean.

Users will have a username and a password for authentication purposes. They will also have their Bilkent id, but as some roles (admin and international office) do not require a Bilkent id, it will be nullable. Users will have personal information (name, surname, email, phone number, their universities), which is a property that they are composed of. They will also have a department property that holds the department's information they are assigned to (department name and department code), which is a property that some roles (such as admin and international office) do not but the others have.

2.1.2.1 Admin

Admin is the user that has the authorization to modify the authentications and authorizations of the other users.

2.1.2.2 International Office

International Office is the role that can receive students' applications to exchange programs and post a score table related to the faculty, in order for the coordinators to place students to the universities.

2.1.2.3 Student

Student is the user role that can apply to Erasmus and Bilateral Exchange programs. Students will be able to create an application if they satisfy the minimum requirements for form creation (exchange type, year and semester to do the exchange), with the ability to modify his/her ongoing applications, such as adding or modifying the university preference list that consists of 5 universities at maximum. Students will also be able to see their list of applications and their status of their applications.

If a student does not like the university he or she is matched with, they can cancel the application, in that case they will be transferred to the waiting bin and students who were not matched with any university will be transferred to the waiting list.

After a student is matched with a university, they will create a pre-approval form. This pre-approval form will include a list of courses the student chose among the offerings of that particular university, the university's name, year and semester of the exchange. The pre-approval form will also have a state (academic coordinator inspection, approved and etcetera). In case of potential corner cases (a course not being offered due to prerequisite related issues and etcetera) the students will be able to create more than one pre-approval forms related to their applications. They will upload the pre-approval form document (in pdf format) to the system, later to be signed by the related coordinator.

If a course is not listed among the approved mandatory courses or if the student wants to take an elective course, he or she will be able to create a course proposal request and send it to the related instructor if it is a mandatory course, else to the coordinator.

The student will be able to check the FAQ page related to their department. If students have questions that are not written at the FAQ page, they will be able to send direct messages to the coordinator(s). The message will be listed in the coordinator's to-do list as a new item.

2.1.2.4 Erasmus Coordinator

Coordinator is the user role that coordinates the exchange programs, matches students with the universities, nominates the students to the universities before mobility, inspect pre-approval forms and approve/decline elective courses offered by students.

Coordinators will be able to see the applications that the students of the same department send. After the students apply to Erasmus or Bilateral Exchange programs -the deadline will be announced by the coordinator- and the international office posts the scores table for the applied students, the coordinators will be able to check the uploaded table. After that, they will match the students with universities by using a priority queue approach.

The coordinators will be able to approve or decline student's applications. If a student's CGPA drops below the required threshold before the mobility, they will get notified and they will have the opportunity to cancel an ongoing application.

The coordinators will also be able to modify the FAQ panel of their departments and will have a to-do list. In their to-do lists; coordinators will be able to see the messages sent to them, the pre-approval forms submitted by the students, notifications about changes in student's applications and changes done in the pre-approval forms. They will also be able to reply to the direct messages.

Before the mobility period, coordinators will be able to see the pre-approval forms the students have submitted and will be able to inspect them within the system or by downloading the documents in order to approve or decline a particular form. They will also be able to write comments based on their inspections and be able to change the status of the pre-approval form later on.

Coordinator will be able to monitor elective course requests sent by the students and approve/decline those course requests. If a course is approved, it will not be added as an approved course to the system since it is an elective course.

They are going to be able to add new universities or modify existing ones with a university request which includes the university's name, the coordinator dealing with the particular university, its quota and some additional information (semester availability, language requirements, etcetera). They will also be able to remove a university from the system.

2.1.2.5 Course Coordinator

Course Coordinator is the user role that can evaluate course proposals of their respective courses. The instructor user will be able to see the previously accepted courses and their corresponding Bilkent course. The course coordinator user will be able to see the proposed courses and the syllabuses of the courses uploaded by the student. After evaluating the proposed course and the uploaded syllabus, the instructor user will be able to decline or accept and add the course to the previously accepted course list. The course coordinator users will also be able to manage their to do lists and get/send notifications.

2.1.2.6 Chair

Chair is the user role that can evaluate the final forms. The chair user will be able to view the applications of the students. The chair user will be also able to accept or reject the final application of a student. Incoming student transcripts from the partner universities will also be evaluated by the chair user. The chair users will also be able to manage their to do lists and get/send notifications.

2.1.2.7 Dean/Director

Dean is the user role that can accept or reject student forms. The dean users will also be able to manage their to do lists and get/send notifications.

2.1.3 Application Forms

Two different forms will exist in the Erasmus application. The first one is the university selection form. In the university selection form the student will select up to 5 partner universities and 1 semester (fall/spring) to apply. Students will finalize their selection forms and they will be matched to a university according to their Erasmus score and their university selections. If a student is matched with a university, they will move on to the second form which is the pre-approval form. In the second form, students will create a list of proposed courses. Students will be able to choose a previously accepted course from their matched university or will be able to propose a new course.

Students will create proposals by uploading the syllabi and the web pages of the courses that they intend to take. Then students will choose the equivalent Bilkent course and submit the proposals. If that proposed course's equivalent Bilkent course is a mandatory course, the system will direct the proposal to the instructor of that course. If the proposed course's equivalent Bilkent course is an elective course, the system will direct the proposal to the Erasmus coordinator. After the proposed courses are approved, students will then prepare the second form with their approved courses and/or the previously approved courses. The second form will first be evaluated by the coordinator and then by the faculty admission board. If any changes occur, the student can resend the pre-approval form.

2.1.4 Features

2.1.4.1 To Do List

In the current status of the Erasmus application process, everything revolves around emails. This can get messy and some of the e-mails can go unnoticed or be forgotten by mistake due to the increasing number of incoming emails. A to-do list which precisely shows all of the incomplete tasks will be provided to the exchange coordinators, chair and dean/director in order to prevent any chance of an erasmus application being overlooked.

2.1.4.2 Calendar

Students usually do not check the relevant information that is presented in webpages. Even if they take a look, they might forget some of the information later such as the due dates. A calendar will be provided to all students so that all students will have an easier access to the knowledge that requires careful attention. Other users will also use the calendar to keep track of the time they have left for evaluating the applications and forms.

2.1.4.3 Messenger

The exchange coordinators and students agree that communication during the Erasmus program applications can get messy. Coordinators have to answer emails from possibly hundreds of students and during this process some might get lost in the commotion and go unanswered. To reduce this mess, our application will give its users the ability to send direct messages to the relevant people. This will reduce the odds that a student's message goes unanswered, and it will also reduce the delay in communication.

2.1.4.4 Application Assistant

In order to help students to prepare more realistic university preference forms, the application assistant will sort the universities according to the likelihood of the student being able to go to that university based on his/her Erasmus score. Furthermore, the application assistant will also help students with their course selections considering the courses that they have already taken and the courses offered by the host university.

2.2 Non-Functional Requirements

2.2.1 Usability

- The website will be easily navigable. Any user will be able to log in and do the required process without any prior knowledge.
- The calendar and to-do list features will allow users to go about their work without the need of reading any guidelines
- Students will be able to prepare their course selection forms without leaving the website

2.2.2 Reliability

- A user's application should not be lost due to any problems.
- The website should have appropriate error messages that show the user what the issue is.

2.2.3 Performance

- The operations a user can perform will be shown clearly. The results of these operations should be shown to the user in less than 3 seconds.
- The load times will be kept below 2 seconds.

2.2.4 Supportability

- The application should support Google Chrome version 107 and above, and Firefox version 102 and above.
- The website design should be consistent across different devices and browsers.

2.2.5 Security

- User information should not be visible to any user that isn't required to see.
- The website should be protected from any possible cyber-attacks.

2.3 Pseudo Requirements

- The application must be a web application.
- The language that will be used for the backend must support object-oriented programming.
- Different types of users must exist with different responsibilities.
- The application must reduce paper usage.
- Students must be able to see the status of their application.
- Coordinators must be able to see a to-do list for waiting items.
- The workload between coordinators must be distributed.
- All approved/rejected courses must be logged and that info must be available to students and coordinators.
- Faculty Administration Board decisions must be logged.
- Changes to pre-application and changes/amendments to the course transfer must be supported.
- All forms must be kept in pdf format and query support must exist.

2.4 System Models

2.4.1 Use Case Model

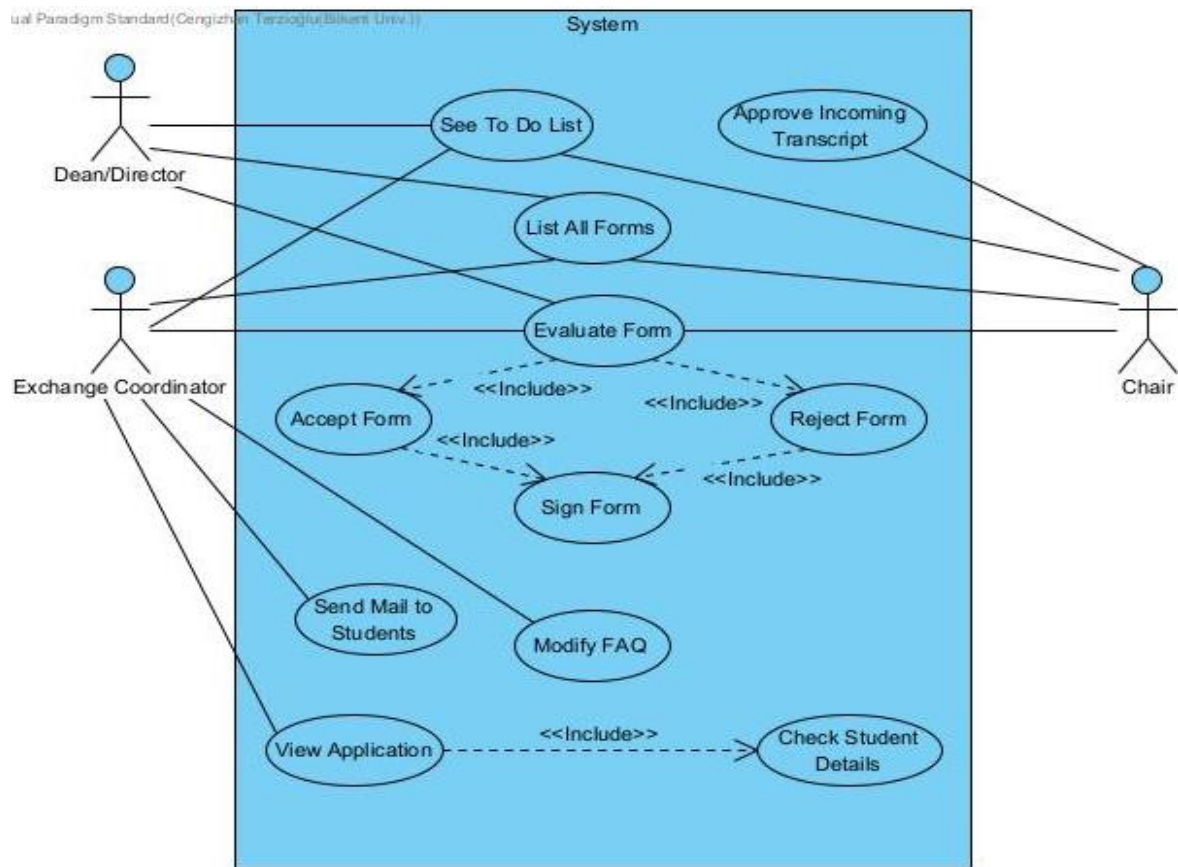


Figure 1: Use Case Diagram for Exchange Coordinator, Chair and Dean/Director

<i>Use Case:</i>	View To-Do List
<i>Participating Actors:</i>	Exchange Coordinator, Chair or Dean/Director
<i>Entry Condition:</i>	The user has some tasks that are incomplete.
<i>Exit Condition:</i>	The user has the list of their tasks.
<i>Flow of Events:</i>	1. The user clicks to view their to do list items.
<i>Quality Requirements:</i>	None

<i>Use Case:</i>	List All Forms
<i>Participating Actors:</i>	Exchange Coordinator, Chair, International Office or Dean/Director
<i>Entry Condition:</i>	None
<i>Exit Condition:</i>	The user has the list of all forms within specified filtering conditions.
<i>Flow of Events:</i>	1. The user selects conditions for filtering. 2. All of the forms that satisfy the conditions that are listed.
<i>Quality Requirements:</i>	Exchange coordinator and chair cannot list the forms of other departments.

<i>Use Case:</i>	Evaluate Form
<i>Participating Actors:</i>	Exchange Coordinator, Chair of Dean/Director
<i>Entry Condition:</i>	The user opens a form to evaluate.
<i>Exit Conditions:</i>	The form is evaluated and a notification is sent to the student.
<i>Flow of Events:</i>	<ol style="list-style-type: none"> 1. The user analyzes the form. 2. The user decides whether to accept or reject the form.
<i>Quality Requirements:</i>	The form has to be in pdf format.

<i>Use Case:</i>	Accept Form
<i>Participating Actors:</i>	Exchange Coordinator, Chair of Dean/Director
<i>Entry Condition:</i>	The user clicks the “Approve” button while evaluating a form.
<i>Exit Conditions:</i>	The form is accepted and its status is updated accordingly.
<i>Flow of Events:</i>	<ol style="list-style-type: none"> 1. Status of the form is updated to “Accepted”. 2. System requests the signature of the user.
<i>Quality Requirements:</i>	None

<i>Use Case:</i>	Reject Form
<i>Participating Actors:</i>	Exchange Coordinator, Chair of Dean/Director
<i>Entry Condition:</i>	The user clicks the “Decline” button while evaluating a form.
<i>Exit Conditions:</i>	The form is declined and its status is updated accordingly.
<i>Flow of Events:</i>	<ol style="list-style-type: none"> 1. Status of the form is updated to “Rejected”. 2. System requests the signature of the user.
<i>Quality Requirements:</i>	None

<i>Use Case:</i>	Sign Form
<i>Participating Actors:</i>	Exchange Coordinator, Chair or Dean/Director
<i>Entry Condition:</i>	The user clicks either to the “Approve” or to the “Decline” button.
<i>Exit Conditions:</i>	The form is signed. A notification is sent to the student in order to report the status of the form.
<i>Flow of Events:</i>	<ol style="list-style-type: none"> 1. The user signs the form. 2. A notification is sent to the student in order to report the status of the form.
<i>Quality Requirements:</i>	None

<i>Use Case:</i>	Modify FAQ
<i>Participating Actor:</i>	Exchange Coordinator
<i>Entry Condition:</i>	None
<i>Exit Condition:</i>	FAQ is updated.
<i>Flow of Events:</i>	<ol style="list-style-type: none"> 1. Exchange coordinator opens the menu for modifying the FAQ. 2. Exchange coordinator adds new questions and answers to the FAQ. 3. Exchange coordinator updates already existing items.
<i>Quality Requirements:</i>	None

<i>Use Case:</i>	Send Mail to Students
<i>Participating Actor:</i>	Exchange Coordinator
<i>Entry Condition:</i>	There should at least be one student who applied to the Erasmus program.
<i>Exit Condition:</i>	Student receives an e-mail from exchange coordinator.
<i>Flow of Events:</i>	<ol style="list-style-type: none"> 1. Exchange coordinator selects the application/form of a student. 2. Exchange coordinator clicks the “Send Mail” button. 3. Exchange coordinator writes and sends an e-mail to the student.
<i>Quality Requirements:</i>	None

<i>Use Case:</i>	View Application
<i>Participating Actor:</i>	Exchange Coordinator
<i>Entry Condition:</i>	There should at least be one student who has applied to the Erasmus program.
<i>Exit Condition:</i>	Exchange coordinator sees the copy of the application made by the selected student.
<i>Flow of Events:</i>	<ol style="list-style-type: none"> 1. Exchange coordinator selects the application of a student.
<i>Quality Requirements:</i>	None

<i>Use Case:</i>	Check Student Details
<i>Participating Actor:</i>	Exchange Coordinator
<i>Entry Condition:</i>	There should be at least one student who applied to Erasmus program.
<i>Exit Condition:</i>	Exchange coordinator sees all the information about the selected student.
<i>Flow of Events:</i>	<ol style="list-style-type: none"> 1. Exchange coordinator clicks the “Check Student Details” button.
<i>Quality Requirements:</i>	None

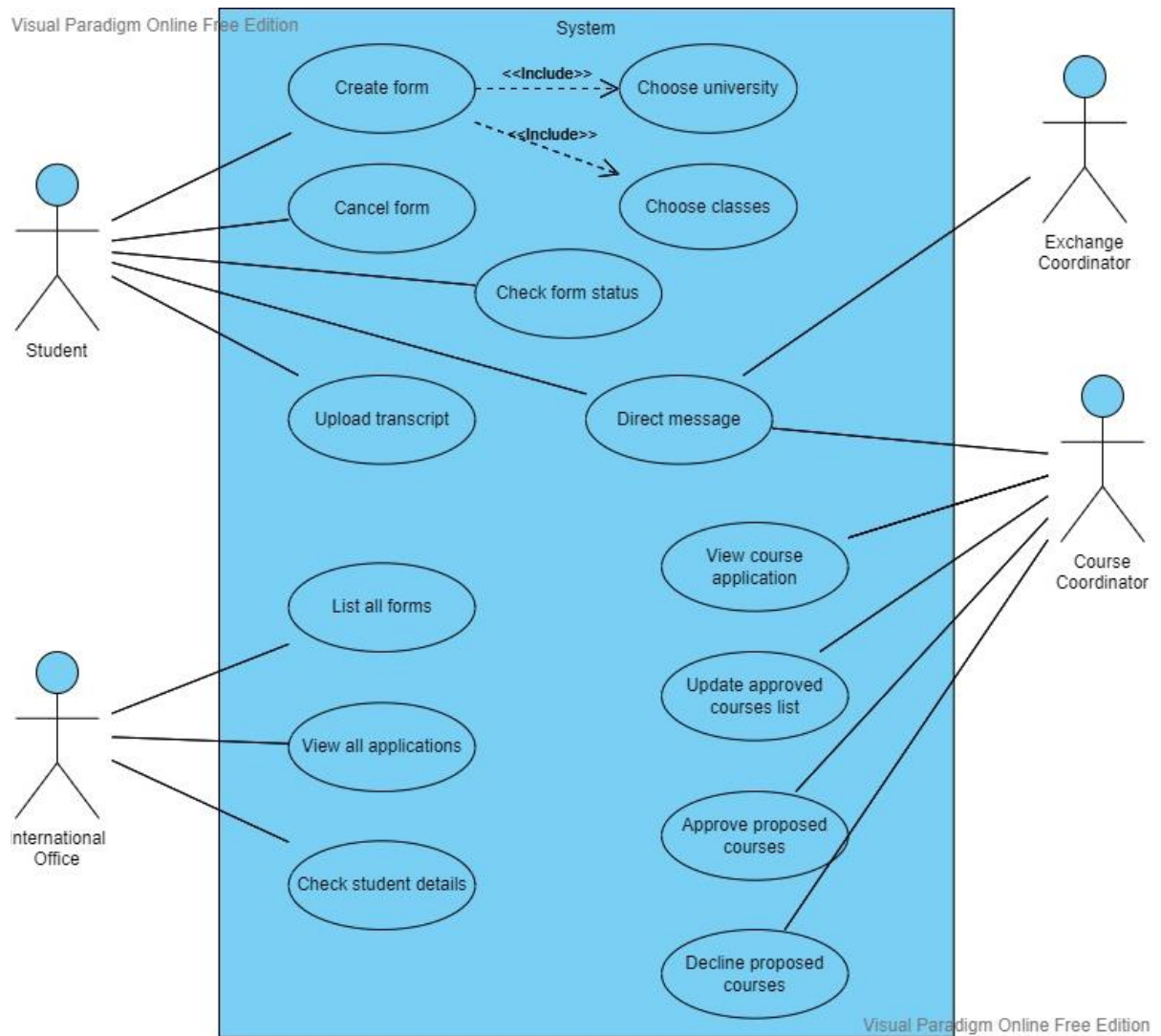


Figure 2: Use Case Diagram for Student, International Office and Course Coordinator

Use Case:	Create form
Participating Actor:	Student
Entry Condition:	User must not have an already existing form.
Exit Condition:	The user fills their form completely
Flow of Events:	1. The student applies to the Erasmus program
Quality Requirements:	The form must be in pdf format.

Use Case:	Choose university
Participating Actor:	Student
Entry Condition:	User must be editing their form
Exit Condition:	The user chooses their university.
Flow of Events:	1. The user accesses a list of universities. 2. The user chooses their 5 wanted universities.
Quality Requirements:	None

<i>Use Case:</i>	Choose classes
<i>Participating Actor:</i>	Student
<i>Entry Condition:</i>	Users must be editing their forms.
<i>Exit Condition:</i>	The user chooses their courses.
<i>Flow of Events:</i>	1. The user accesses the list of available classes. 2. The user chooses their wanted classes.
<i>Quality Requirements:</i>	None
<i>Use Case:</i>	Cancel form
<i>Participating Actor:</i>	Student
<i>Entry Condition:</i>	Users must already have an existing form.
<i>Exit Condition:</i>	The user deletes their form
<i>Flow of Events:</i>	1. The form is removed from the database.
<i>Quality Requirements:</i>	None
<i>Use Case:</i>	Check form status
<i>Participating Actor:</i>	Student
<i>Entry Condition:</i>	Users must already have an existing form.
<i>Exit Condition:</i>	The user accesses their form.
<i>Flow of Events:</i>	1. The student sees their form's application status.
<i>Quality Requirements:</i>	None
<i>Use Case:</i>	Upload transcript
<i>Participating Actor:</i>	Student
<i>Entry Condition:</i>	Users must have their Erasmus program completed.
<i>Exit Condition:</i>	The user submits their transcript.
<i>Flow of Events:</i>	1. Students choose their transcript file and upload it.
<i>Quality Requirements:</i>	The transcript must be in pdf format.
<i>Use Case:</i>	Direct message
<i>Participating Actors:</i>	Student, Exchange Coordinator, Course Coordinator
<i>Entry Condition:</i>	None.
<i>Exit Condition:</i>	The user sends their message.
<i>Flow of Events:</i>	1. The user selects who they want to message. 2. The user sends a message.
<i>Quality Requirements:</i>	None
<i>Use Case:</i>	List all forms
<i>Participating Actor:</i>	International Office
<i>Entry Condition:</i>	The user must have forms submitted to them.
<i>Exit Condition:</i>	The user has the list of all forms.
<i>Flow of Events:</i>	1. All forms that satisfy the query conditions are listed.
<i>Quality Requirements:</i>	None

Use Case:	View all applications
Participating Actor:	International Office
Entry Condition:	The user must have applications submitted to them.
Exit Condition:	The user has the list of all applications.
Flow of Events:	1. The user selects an application. 2. The application chosen is viewed.
Quality Requirements:	None

Use Case:	Check student details.
Participating Actor:	International Office
Entry Condition:	None.
Exit Condition:	The user accesses the information of the student.
Flow of Events:	1. The user selects a student. 2. The student's information is viewed.
Quality Requirements:	None

Use Case:	View course application
Participating Actor:	Course Coordinator
Entry Condition:	There must be courses applied to.
Exit condition:	The user sees the courses applied.
Flow of Events:	1. The user selects and views the application
Quality Requirements:	None

Use Case:	Update approved courses list
Participating Actor:	Course Coordinator
Entry condition:	None.
Exit Condition:	The user updates the course list.
Flow of Events:	1. The list of approved courses is updated.
Quality Requirements:	None

Use Case:	Approve proposed courses.
Participating Actor:	Course Coordinator
Entry Condition:	There must be a course that has been applied to.
Exit condition:	The user approves the course application.
Flow of Events:	1. The course is added to the approved courses list.
Quality requirements:	None

Use Case:	Decline proposed courses
Participating Actor:	Course Coordinator
Entry Condition:	There must be a course that has been applied to.
Exit Condition:	The user denies the course application.
Flow of Events:	1. The course application is denied.
Quality Requirements:	None

2.4.2 Dynamic Models

2.4.2.1 State Diagrams

2.4.2.1.1 Student State Diagram

Visual Paradigm Standard (Murat Güney Kemal (Bilkent Univ.))

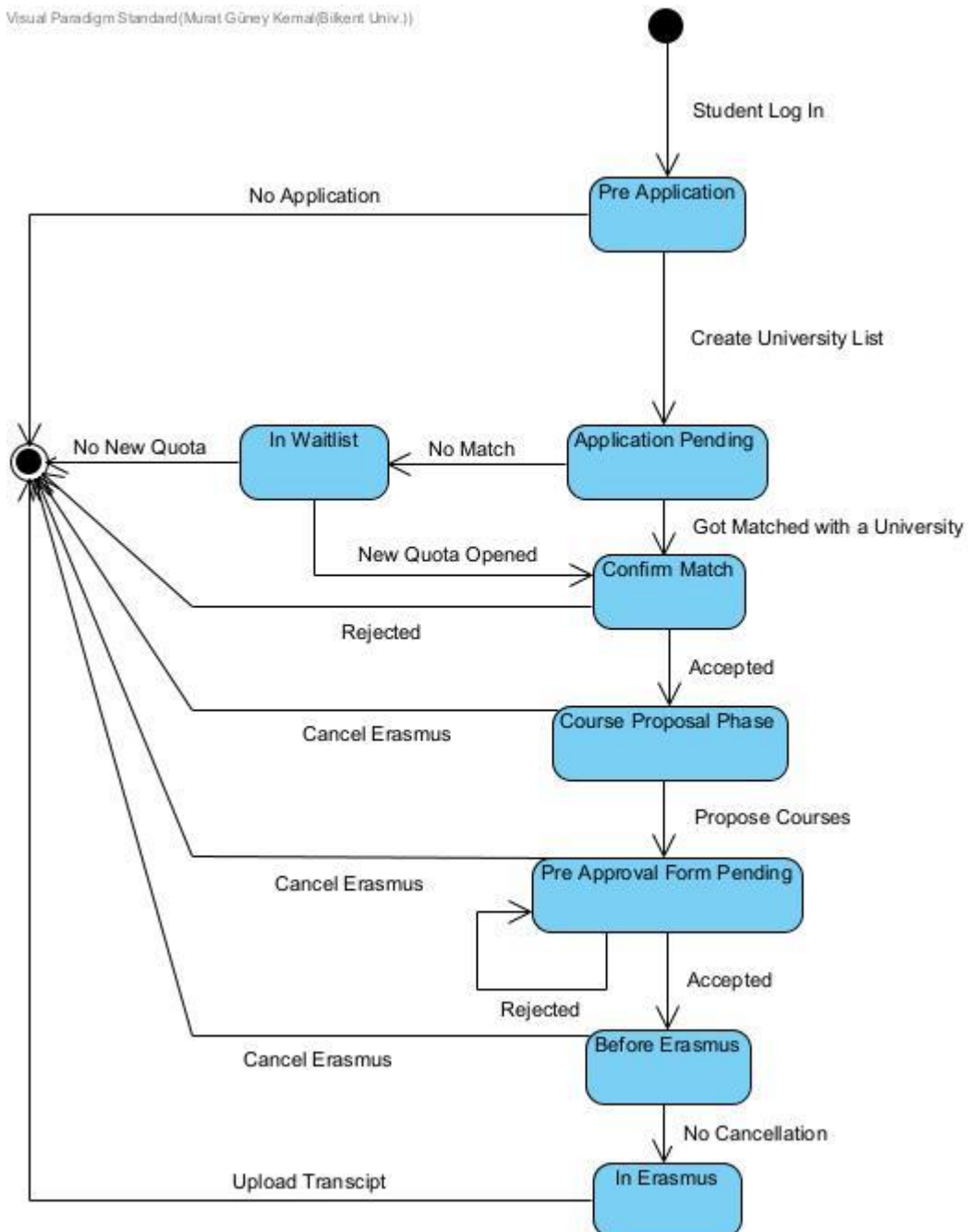


Figure 3: The student state diagram mimics the whole Erasmus application process. The student changes states according to their stage in the Erasmus application.

2.4.2.1.2 Application State Diagram

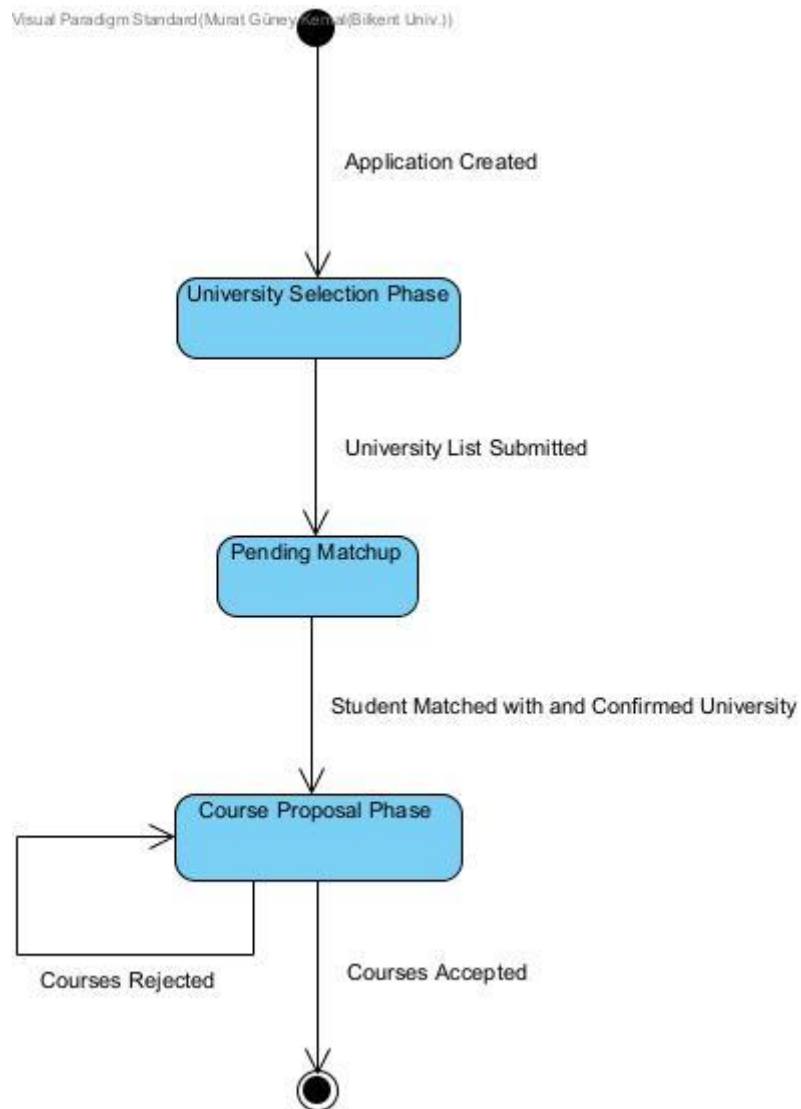


Figure 4: The application state diagram shows the collective processes of all forms that students submit for Erasmus applications. The application changes state according to the submission or the evaluation of each form.

2.4.2.1.3 University State Diagram

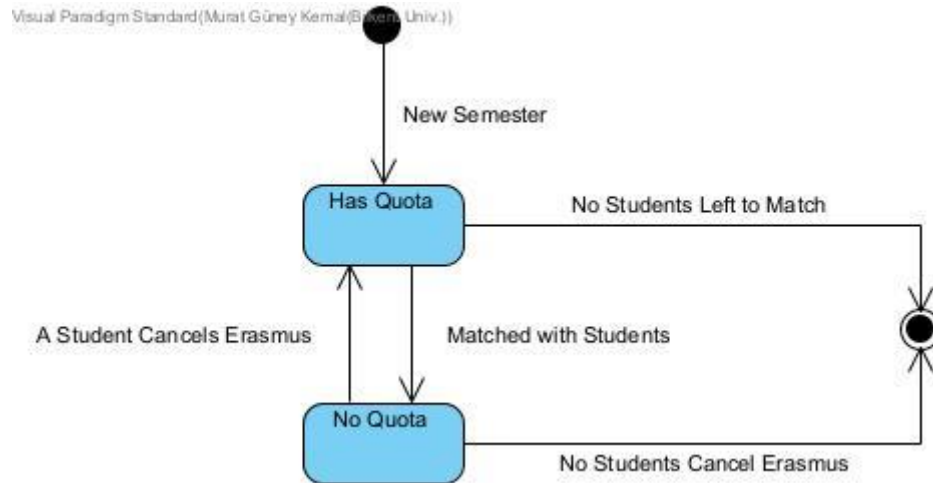


Figure 5: The university state diagram shows the process of matching universities with students. The university changes state according to its quota and unmatched students.

2.4.2.2 Activity State Diagram

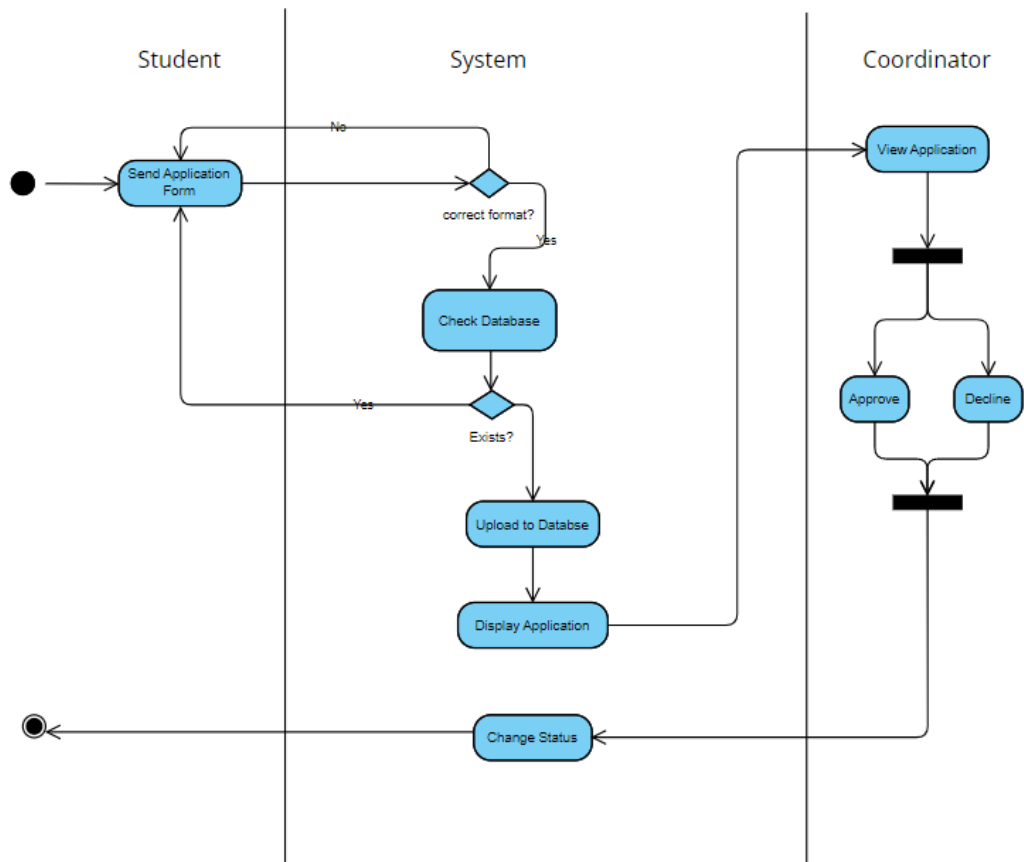


Figure 6: Activity State Diagram

2.4.2.3 Sequence Diagram

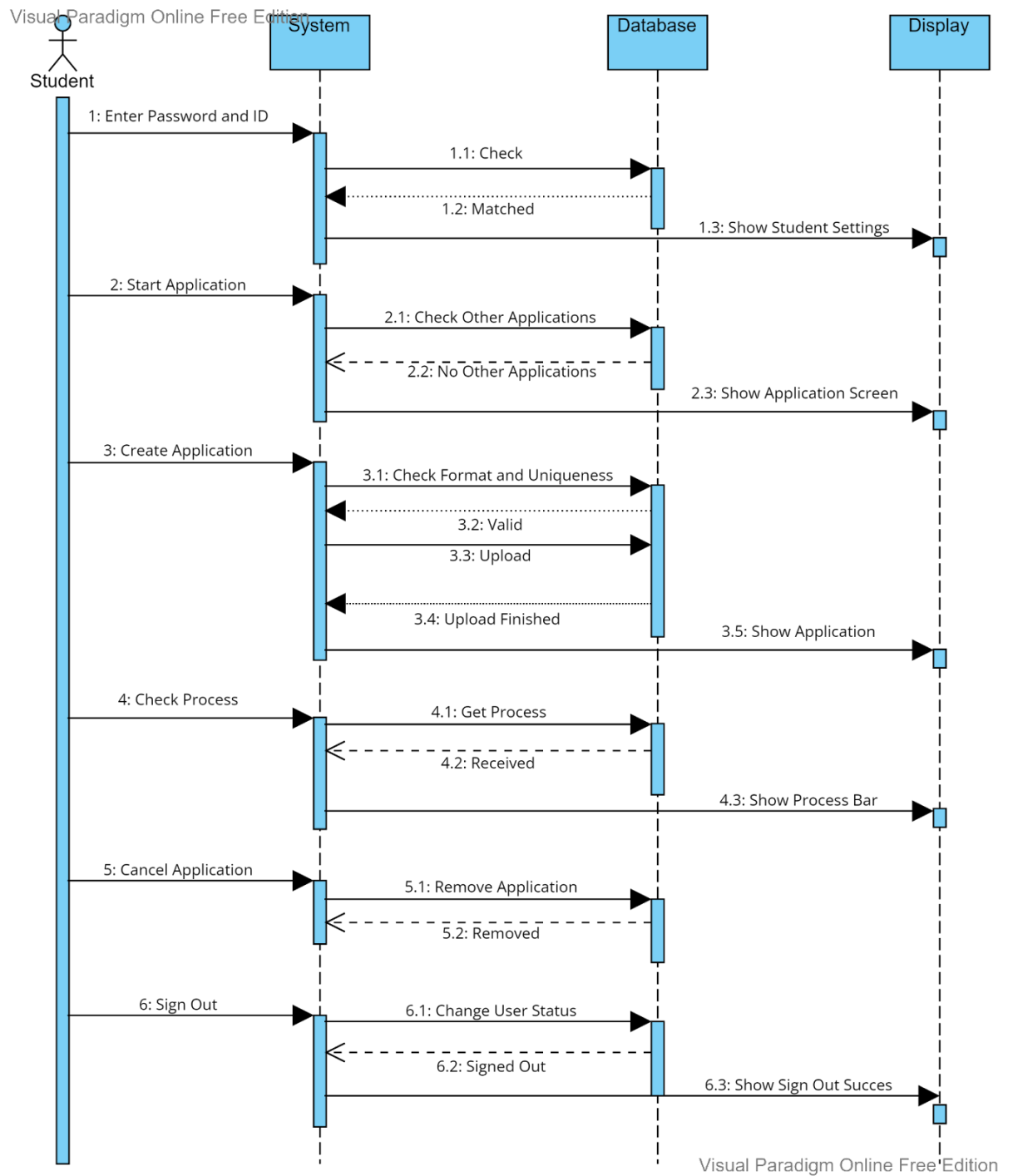


Figure 7: Sequence Diagram

Visual Paradigm Online Academic Partner Program



```
classDiagram
    class Application {
        - id: long
        - student: Student
        - status: Status
        - universityQueue: UniversityQueue
        - preapprovalForm: PreapprovalForm
        - studentCgpa: double
        - type: int
        - academic_year: string
        - semester: string
        - assigned_university: University
    }
    class UniversityQueue {
        - id: long
        - application_id: long
        - university_queue: List<University>
    }
    class University {
        - id: long
        - name: string
        - courses_taught_in_eng: string
        - fall_semester_availability: bool
        - spring_semester_availability: bool
        - quota: string
        - language_requirements_eng: string
        - language_requirements_other: string
        - coordinator: Coordinator
        - exchange_type: int
    }
    class PreapprovalForm {
        - id: long
        - course_list: List<Course>
        - preapproval_document: File
        - status: bool
        - status_comment: string
        - academic_year: string
        - semester: string
        - application_id: long
    }
    class Course {
        - id: long
        - course_code: string
        - university: University
        - course_name: string
        - is_approved: bool
        - has_project: bool
        - ects_credit: double
        - blkent_credit: int
    }
    Application "1" -- "1..5" UniversityQueue
    Application "1" -- "1" University
    Application "1" -- "1" PreapprovalForm
    UniversityQueue "1" -- "1..5" University
    PreapprovalForm "1" -- "1" Course
```

Visual Paradigm Online Academic Partner Program

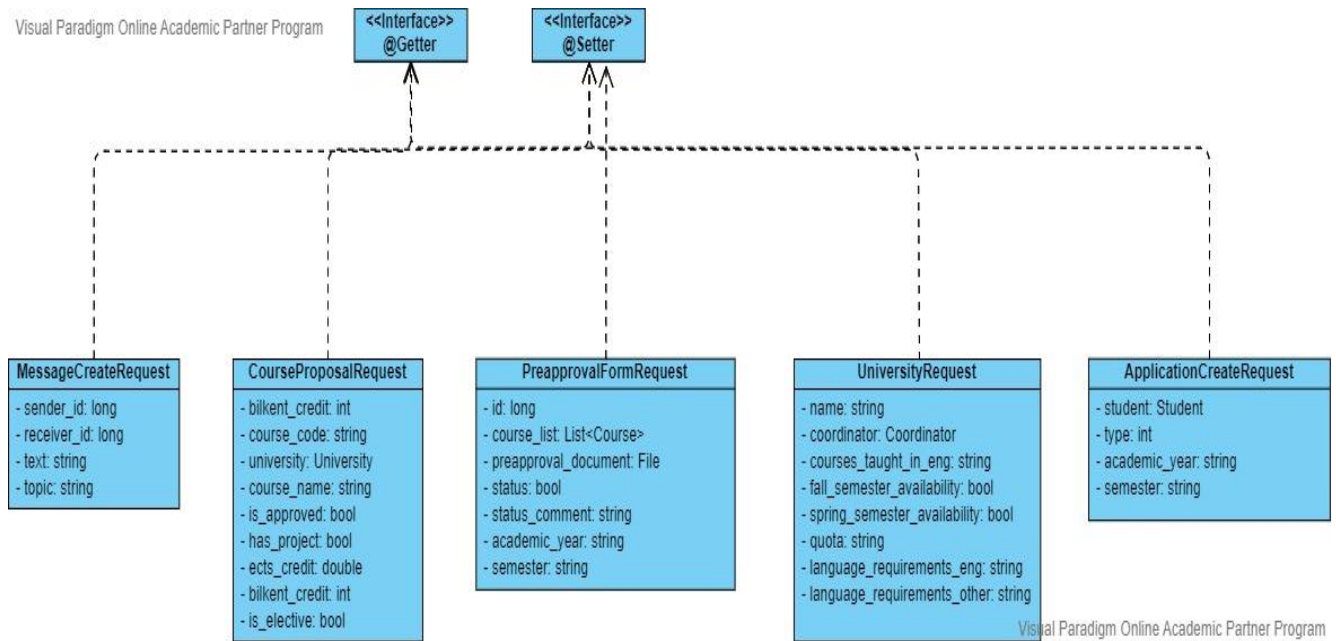


Figure 11: Request Class Diagram

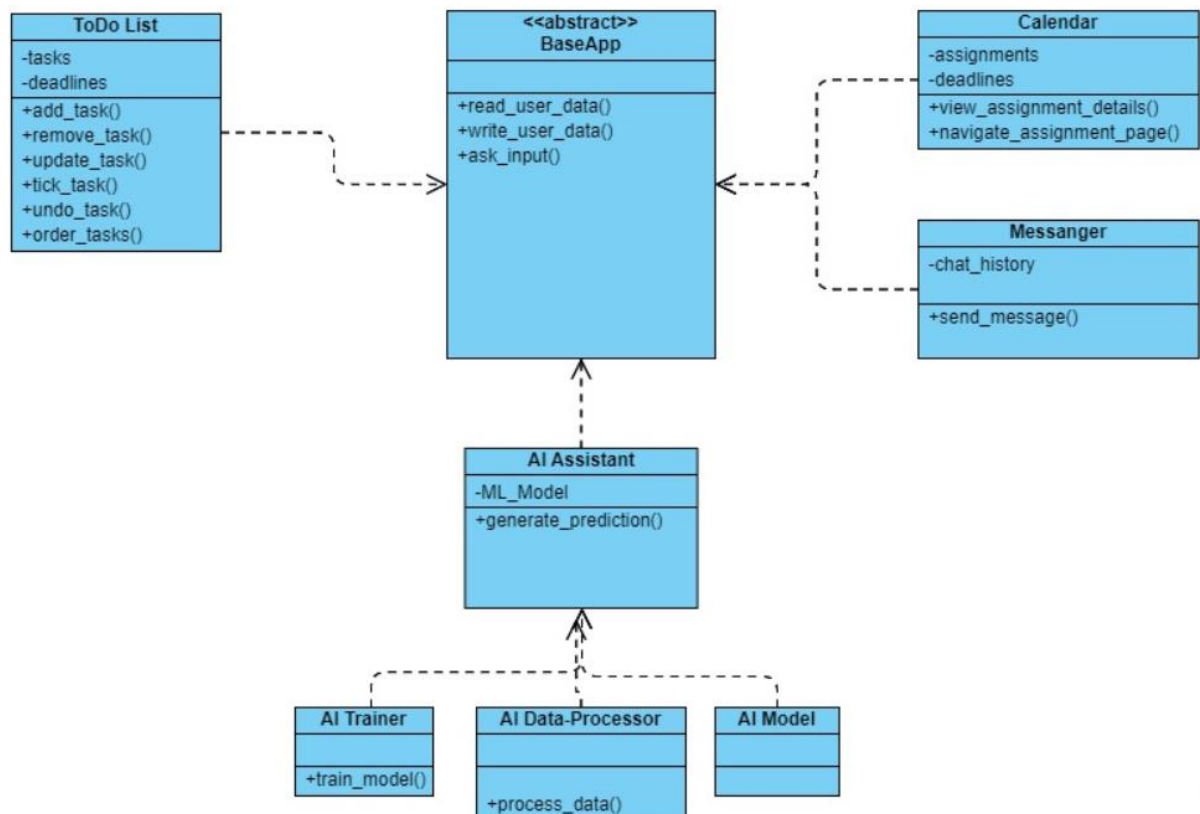


Figure 12: Features Class Diagram

2.4.4 User Interface

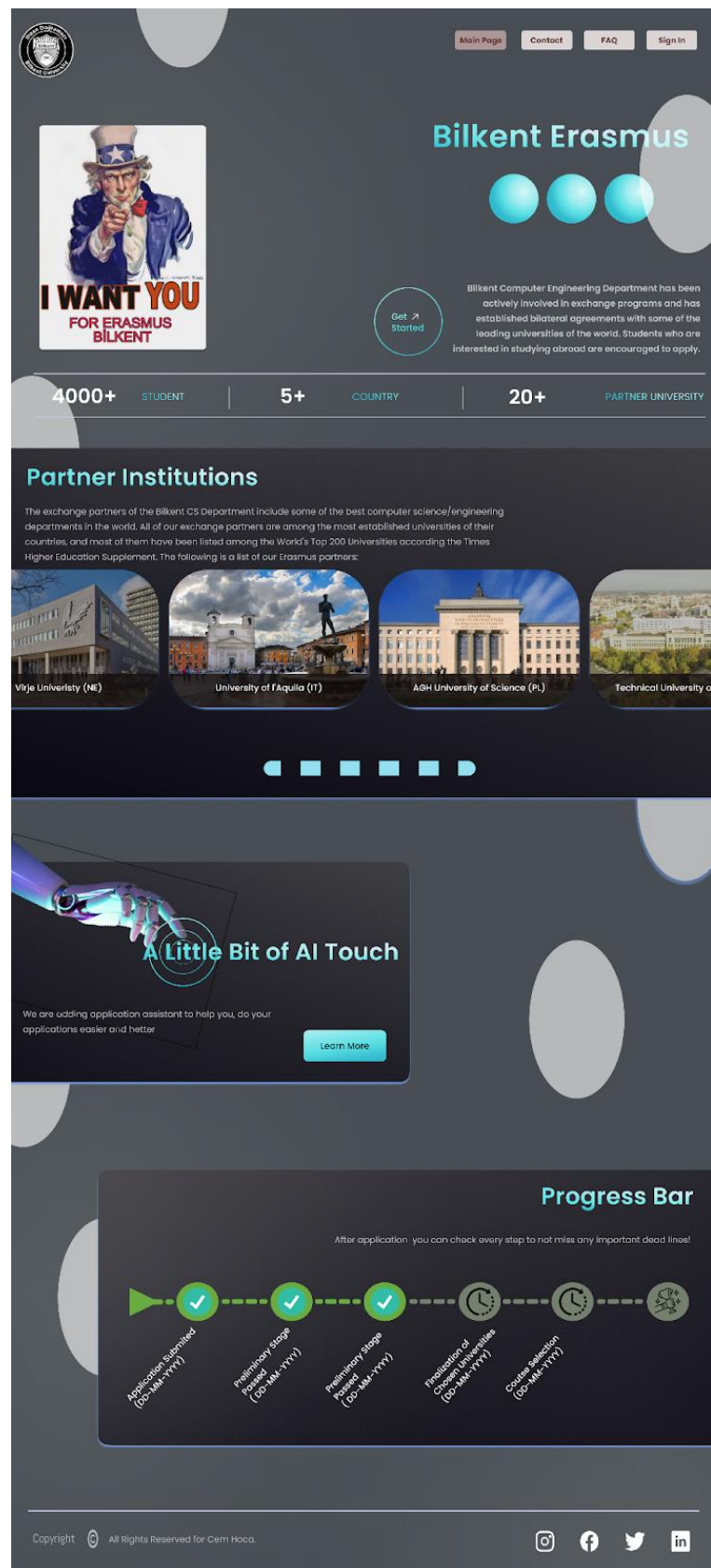



Figure 13: Home Page



Bilkent Erasmus

[Main Page](#)[Contact](#)[FAQ](#)[Sign In](#)

Contact

Department Coordinators

Can Alkan <http://cs.bilkent.edu.tr/~calkan/>
Ayşegül Dündar <http://www.cs.bilkent.edu.tr/~adundar/>

Administrative Coordinators

Deniz Dicle Arman
Erkin Tarhan terkin@bilkent.edu.tr

Erasmus Students Counselors

Elif Ünsal elif.unsal@bilkent.edu.tr
Buket HATAM buketg@bilkent.edu.tr


Bilkent ID

Password

Sign In

Forget Password?


Figure 14: Login Page



Bilkent Erasmus

[Main Page](#)[Contact](#)[FAQ](#)[Sign In](#)

Frequently Asked Questions



Q.1. What is the student exchange program?

×

An important component of university education is to develop a better understanding of the world, not just academically but also socially. Student exchange is a means to achieve this goal by enabling students study at an environment different from their home country and institution.

Q.2. What are the financial issues involved for an exchange student?

+

Q.3. Is housing provided to exchange students at the host universities?

+

Q.4. Who can apply for the exchange program?

+

Q.5. Can't I go to exchange on the 4th year?

+

Q.6. How can I apply for the exchange program?

+

Q.7. Is there a GPA requirement for exchange applications?

+

Figure 15: FAQ Page

23

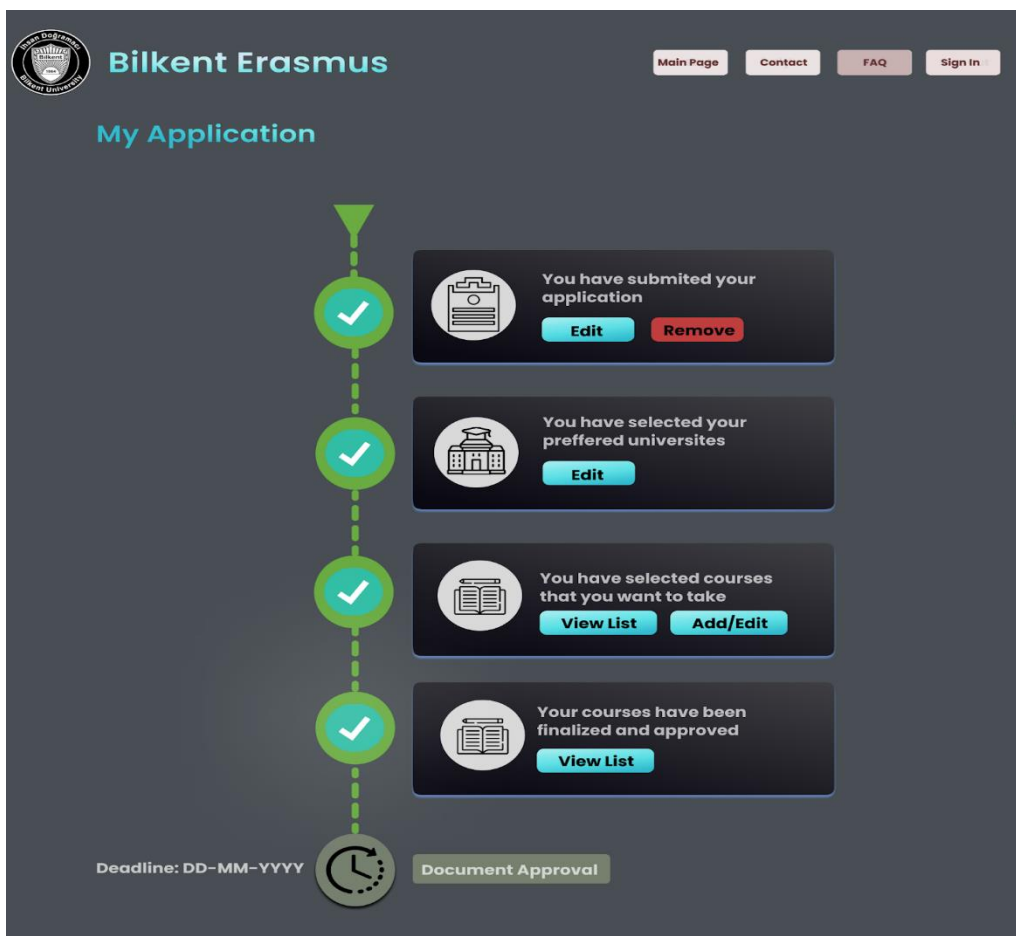
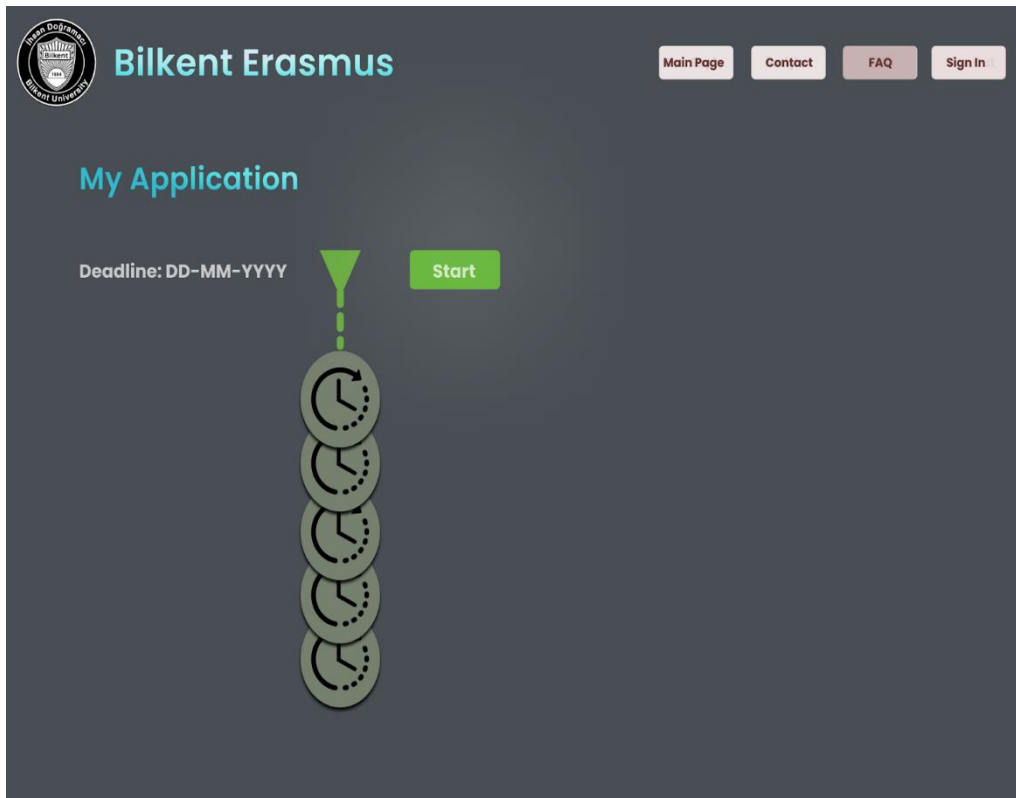


Figure 16: Progress Bars

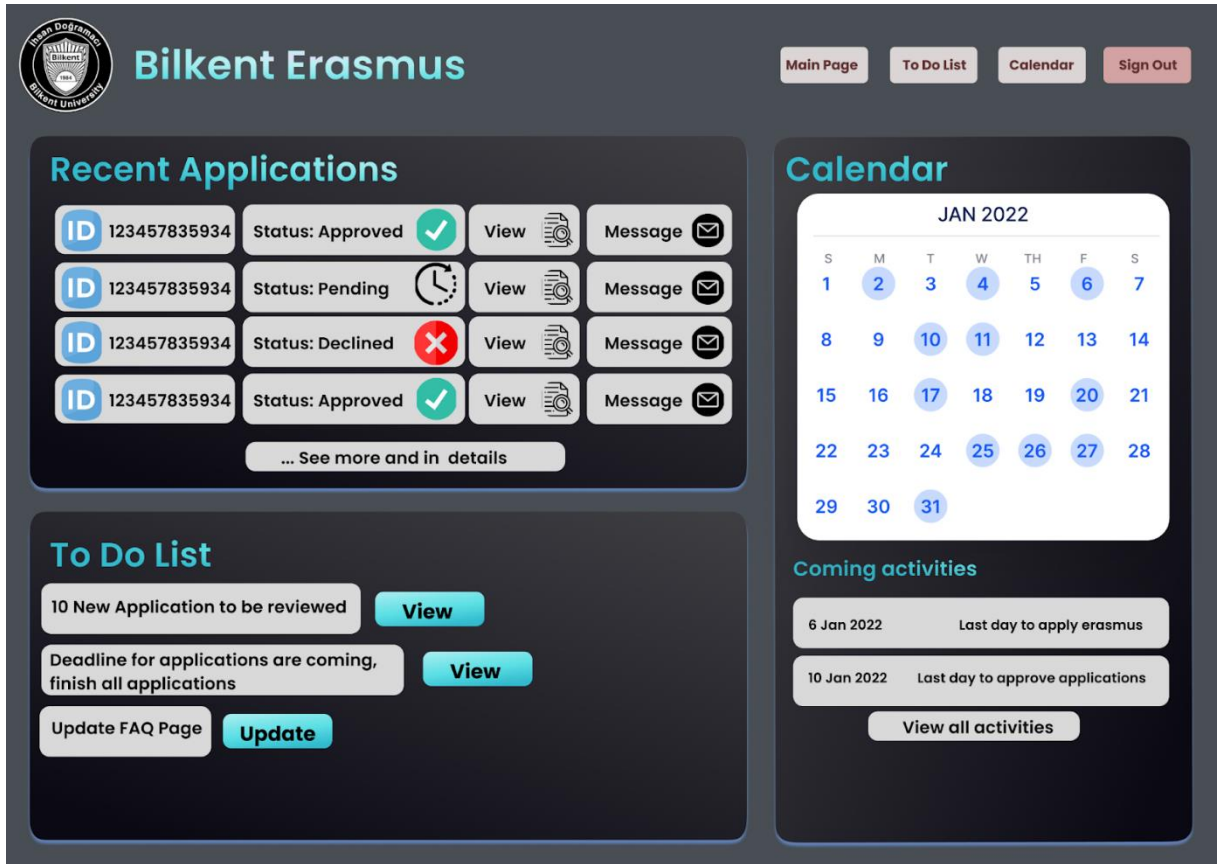


Figure 17: Features Page

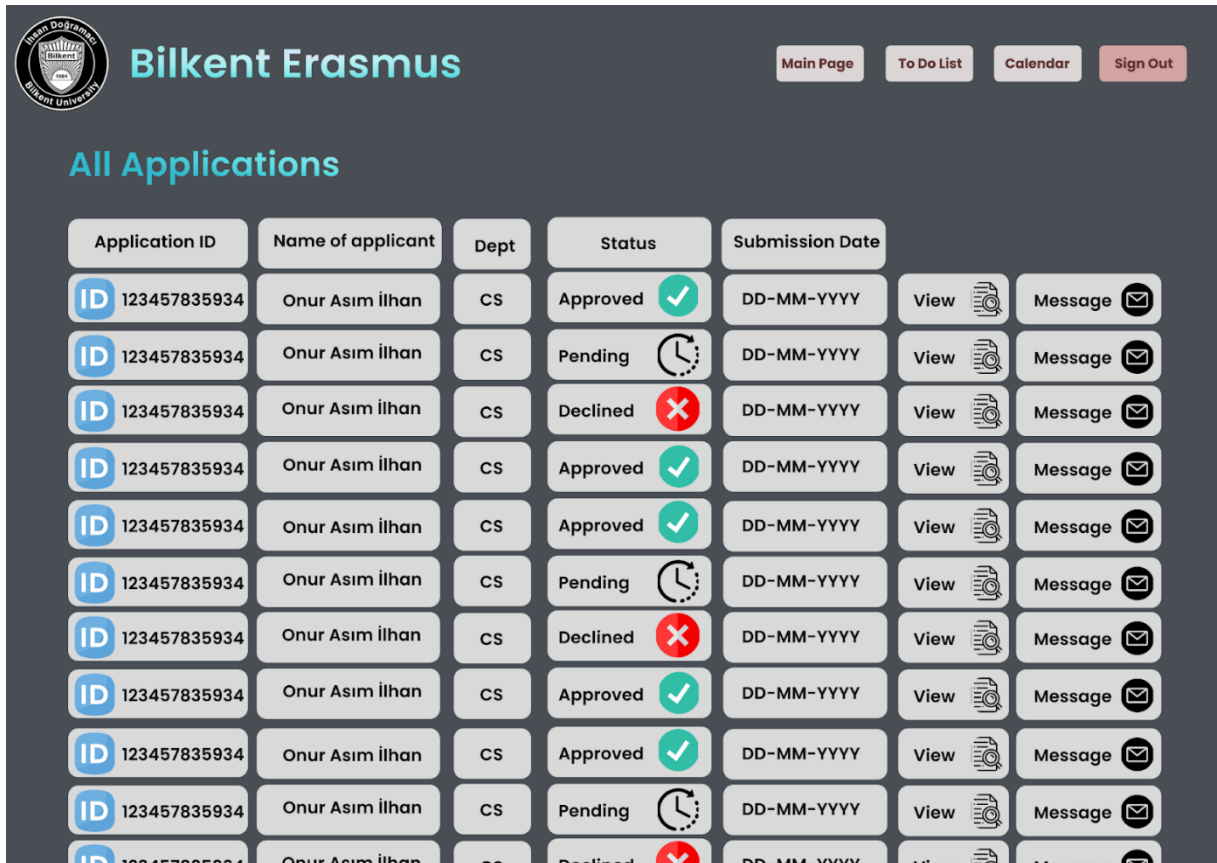




Figure 18: Applications Page




Bilkent Erasmus

[Main Page](#)
[To Do List](#)
[Calendar](#)
[Sign Out](#)

Approved Course List



BILKENT



XYZ University

Choose

CS Courses	
CS319 (description.....)	Syllabus
CS319 (description.....)	Syllabus
CS319 (description.....)	Syllabus
CS319 (description.....)	Syllabus
CS319 (description.....)	Syllabus
CS319 (description.....)	Syllabus
CS319 (description.....)	Syllabus

EE Courses
ME Courses
MATH Courses
MATH Courses
MATH Courses

CS Courses	
CS319 (description.....)	Syllabus
CS319 (description.....)	Syllabus
CS319 (description.....)	Syllabus
CS319 (description.....)	Syllabus
CS319 (description.....)	Syllabus
CS319 (description.....)	Syllabus
CS319 (description.....)	Syllabus
CS319 (description.....)	Syllabus

EE Courses	+
ME Courses	+
MATH Courses	+
MATH Courses	+
MATH Courses	+

Figure 19: Course Selection Page