

Bilkent University
Department of Computer
Engineering



CS319 Term Project

Section 3
Group 3D
Erasmus++

Project Analysis Report

Group Members

- Yunus Eren Türkeri (22001842)
- Ender Utlu (22001983)
- Ege Ayan (22002478)
- Alp Ertan (22003912)
- Mehmet Feyyaz Küçük (22003550)

1. Introduction	4
1.1 Application Features	4
2. Proposed System	5
2.1 Functional Requirements	5
2.1.1 Login Page	5
2.1.2 Profile Page	5
2.1.3 My Applications	5
2.1.4 To-Do List	6
2.1.5 Roadmap	6
2.1.6 Announcements	6
2.2 Non-functional Requirements	6
2.2.1 User Friendly	6
2.2.2 Maintainable	6
2.2.3 Secure	7
2.2.4 Availability	7
3.1 Activity Diagrams	8
3.2 Class Diagram	11
3.3 Sequence Diagram	14
State Diagrams	14
3.4 State Diagrams	15
3.4.2 Placements State Diagram	16
3.4.3 Course Selection State Diagram	17
3.4.4 Mobility Process State Diagram	18
3.4.5 Course Transfer State Diagram	19
3.5 Use Case Diagram	20
4.0 User Interface	21
4.1 Login Screen	21
4.2 “Forgot your password?” Screen	22
4.3 Main Screen (For candidates)	23
4.4 Main Screen (For coordinators)	24
4.5 Profile Screen	25
4.6 Change Password Screen	26
4.7 Post Announcement	27
4.8 Road Map	28
4.10 My Application View	30
4.11 Application Form Screen (Edit)	31
4.12 Application Form Screen (View)	32

4.13 Course Registration Screen	33
4.14 My Applications	34

1. Introduction

As group 3D, our aim is to build a thorough system for Erasmus+ Mobility Program application. It serves as a platform where the candidates (students/applicants) may apply to mobility programs with the eased process steps. Though the candidates are the main audience for application, coordinators, departmental exchange coordinators, administrators will also be able to use this system that will ensure practicality for the application process.

The main objectives of the application are posting announcements, viewing the roadmap and easy application process to Erasmus+ programs. These features focus on making the application process and its management easier for both candidate and coordinator sides. The main feature for this *ease-of-use* is that the interaction between the course coordinators and the candidates are much faster such that not all the process steps are on e-mail or paper, rather the use of paper and e-mails are aimed to be decreased with the aforementioned system's features.

1.1 Application Features

- Profile Management including customization of password, profile picture, email, phone number, date of birth, gender, about me)
- Uploading Application Documents (and required documents such as Learning Agreement, Pre-approval Form, Official Dates Form, etc.)
- Course Selection
- To-Do List coordinators
- Road-map for candidates
- Posting announcements (administrators, coordinators will post important announcements through the system). Approved/rejected courses will be logged here for further students and coordinators to see.
- Faculty Administration Board decisions will be logged (for both-pre approval and credit transfer).

2. Proposed System

2.1 Functional Requirements

The program has three types of users which are

- Candidates
- Coordinators
- Administrators

2.1.1 Login Page

Login page will consist of an email field and a password field for users to enter their login information along with a login button which will terminate the login operation with given parameters (email and password). There will also be a forget password button in case if the user has forgotten his/her password. There will be different types of accounts depending on their level of access such as candidate account, department coordinator account, and admin account.

2.1.2 Profile Page

The profile will contain email, name, phone number, date of birth, gender, about me, and etc. Some of this information will be filled out for the users. But the users need to fill out the rest. Users will be able to update their profile picture and password.

2.1.3 My Applications

If the users are eligible for Erasmus (CGPA ≥ 2.5), they will be able to apply to any of these programs during the application period. Users will have to fill in the necessary boxes including credentials and choice of universities for the application form. Automatic application form will be formed inside the system via fetching the candidate data in the pre-filled boxes. The students will be able to request the ready application form by pressing the “export pdf” button. Statement of Purpose For Erasmus file can be uploaded as pdf format to the application system. Candidates who completed their application will be added to the respective department coordinator’s TODO List as “Approved”. After that, the candidate can continue with the rest of their application.

2.1.4 To-Do List

To- Do list is only visible to department coordinators. To- Do list will make it easier for the coordinator to see at which stage the candidate is at. They will be able to perform various actions depending on the candidate's stage.

2.1.5 Roadmap

Since the applying phase can be complicated, each candidate will have a roadmap to guide them through that application. When a candidate applies for a program, the respective coordinator will be able to view their roadmap. Roadmap might contain steps such as "Fill out the application form", "Upload a Statement of Purpose", etc.

2.1.6 Announcements

One of the biggest problems of the current Erasmus system is lack of communication between candidates and coordinators. Since it can be tedious for coordinators to communicate with every candidate, coordinators can post important announcements that will be visible to candidates. Announcements will also be sent to candidates via noreply mail.

2.2 Non-functional Requirements

2.2.1 User Friendly

Erasmus++ will have informative navigation bars that won't confuse the users. The system will also have query support, and a FAQ section. Also, it is important for the system UI to seem appealing to the users, for example, sharp contrast colors will not be chosen for the user interface. Furthermore, the system requires paper usage only when there is a sign of a user is needed; other fundamental processes won't require any paperwork.

2.2.2 Maintainable

The contemporary libraries and software languages will be used for the implementation, i.e. React v18.0, so that the maintenance for further development would be more efficient. Also, the concept of object-oriented software engineering will be adopted as much as possible, to fix any occasional bugs and necessary updates.

2.2.3 Secure

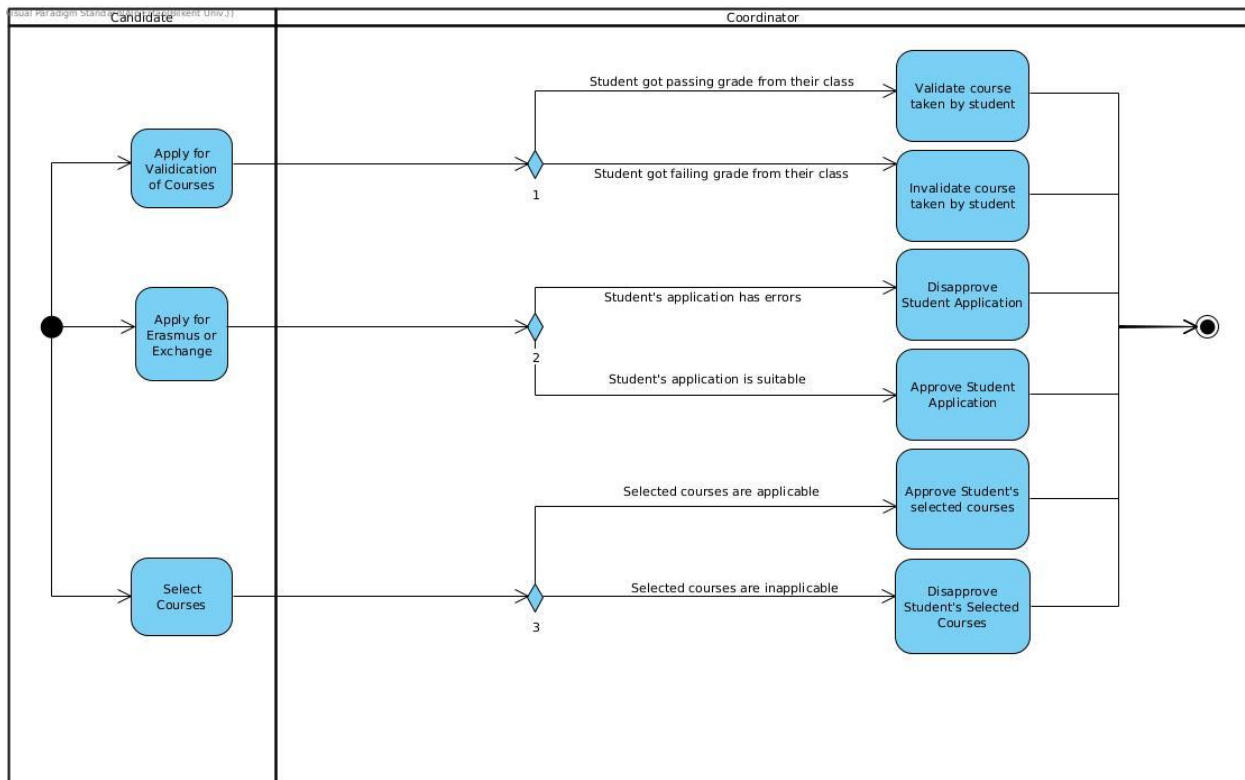
Admins will provide coordinators their own dedicated accounts & passwords at the beginning of each academic semester. This will ensure that no unauthorized user can log in to the system as a coordinator. Given that, the candidate accounts will also be created by the coordinators which ensures that anybody out of the domain will not have any access to the system.

2.2.4 Availability

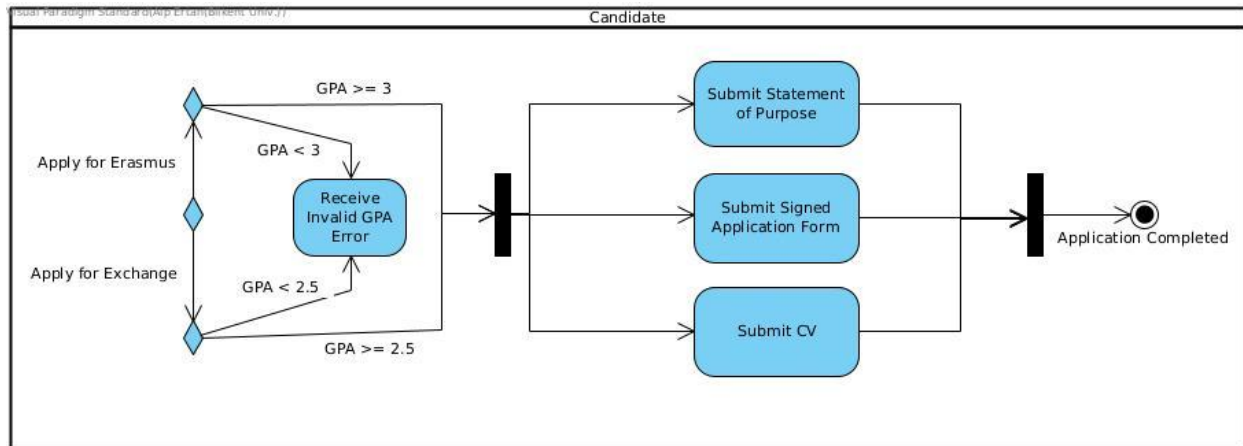
The system will be open for users at a given point in time, unless a pre-announced maintenance break is needed. Note that this does not imply that the candidates will be able to apply mobility programs any time they wish to, the system availability is only for accessing and using the web-application.

3.0 Dynamic Models

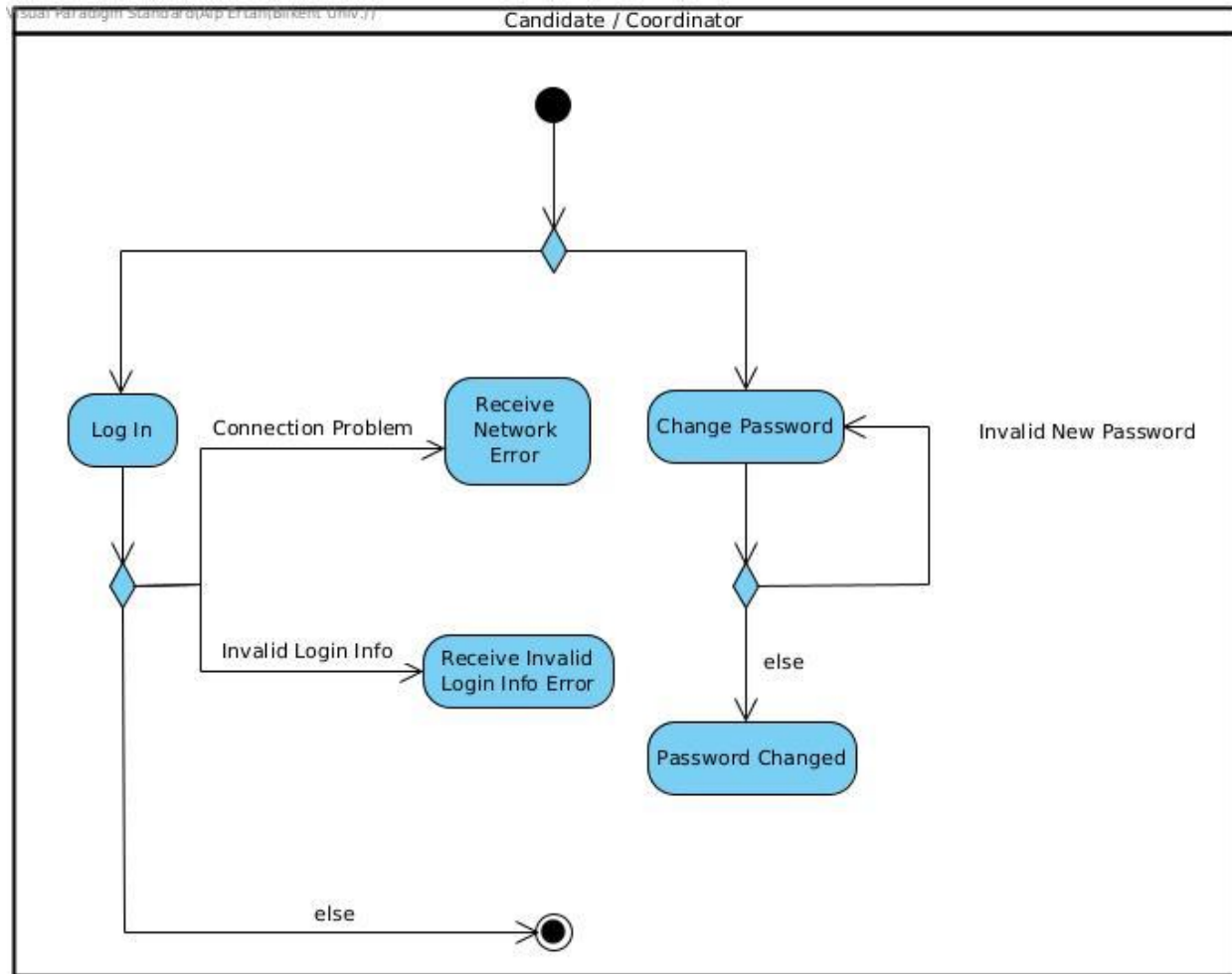
3.1 Activity Diagrams



The dynamic between the candidate and the coordinator is based on three types of requests which are validation of courses, applying for Erasmus program and selecting courses. For course validation, the validation of the taken course will be done by the coordinator depending on if the student gets the passing grade or not. For the Erasmus application case, the two outcomes depend on if the application is valid or not. By investigating the application in terms of errors, the approval statement will be done on the coordinator side. Finally, for the course selection, the coordinator will be deciding if the selected courses are appropriate for the application or not and finally the request will be approved or disapproved.

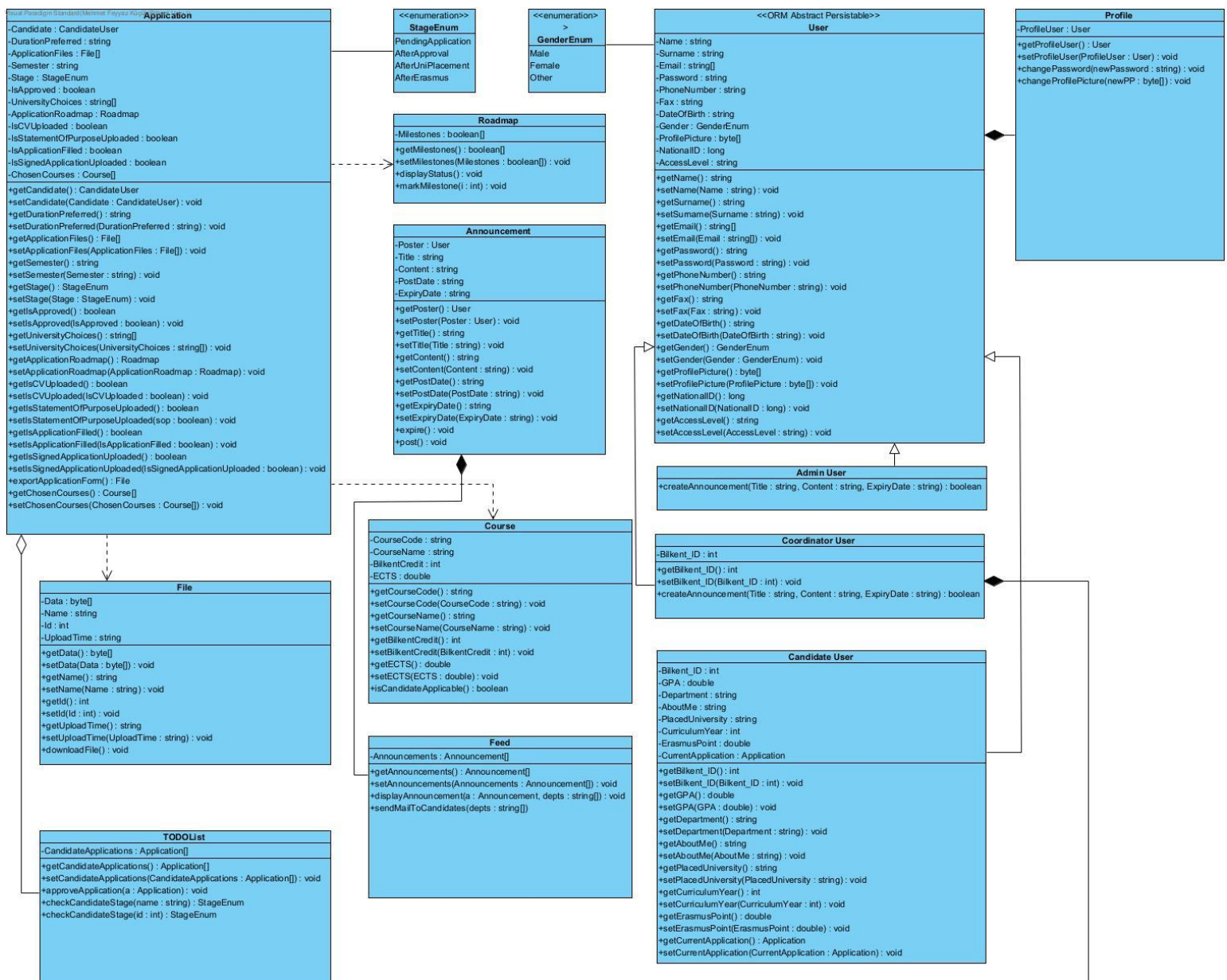


For the erasmus application, the first stage is to determine if the candidate has enough GPA (≥ 2.5) in order to move to further stages. If the candidate fails to fulfill the GPA requirement, he/she will receive an invalid GPA error immediately. If the candidate has the required GPA limit, the candidate will have to submit his/her statement of purpose, application form and CV. In order to complete the application, all three of these forms must be uploaded to the system.



For the login and change password operations, both candidates and coordinators share the same activity diagram since the process is similar. If the user attempts to login, the attempt will either be successful or the user will face a login error. The first type of login error the user might face is the invalid login info such as username / password mismatch. The second type of login error is due to connection error in cases where the server is down or there is no internet connection. For the change password operation, the outcome will be decided on if the new password is invalid or not. If it is valid, the password change operation will be done successfully, if it is invalid, a valid new password will be requested from the candidate/coordinator.

3.2 Class Diagram



Above is the class diagram of Erasmus++ web-application. There are 13 classes in total to model keys, actors, events and functionalities in our applications.

User Class: This class is abstract, and used to model actors in a general form. CandidateUser, CoordinatorUser, AdminUser all inherit from this class to represent a specific actor in our design. The user class contains the credentials of the user.

GenderEnum Class: This is an enumerator for depicting the user's gender. More choices can be added if needed.

CandidateUser Class: This class represents the candidate actor (the user who will apply for Erasmus). Aside from the information coming from the abstract user class, CandidateUser class contains academic information such as ID, GPA, department, curriculum year, about me, and also the Erasmus application as well.

CoordinatorUser Class: This class represents the coordinator actor responsible for the candidate. Aside from the information from the user class, coordinators also have ID and department. Coordinators can also create and post announcements for candidates associated with the coordinator's department.

AdminUser Class: This class represents the admin actor who will regulate and maintain the Erasmus++ application. The AdminUser class can also create and post announcements for everyone.

Profile Class: This class represents the profile of the user. All the elements that are needed to display the profile page will be packed into this class. The Profile class also contains the functionalities changePassword, and changeProfilePicture.

Announcement Class: This class encapsulates the announcement that can be created and posted by coordinators and admins. Poster is the User who created the post. Announcement contains Title and Content inputs which need to be given by the creator of that announcement. PostDate is the time and date the announcement is published. ExpiryDate is the time and date the announcement will be removed or lose its importance when it is no longer relevant.

Feed Class: This class holds and displays the announcements to the relevant candidates. The Relevancy is determined by matching the department of the candidates and coordinators. Admins will create announcements for everyone (including coordinators).

File Class: File class is a template suitable for all file types we will store(pdf, txt, word etc.). Since we can store every file as a byte[], this class can represent all of our files. It has a name, upload time and a unique id. Its downloadFile function allows us to fetch it when we need to.

Stage Enum Class: It has 4 enumerator values for 4 different stages. It is used in all operations that need a stage info. The stages are numbered according to their real life order for logical correlation.

Roadmap Class: Roadmap class represents the milestones of the application process. Its milestones attribute stores the complete and uncomplete milestones in it. It is used by application classes to keep track of milestones. The markMilestone function marks the ith milestone as complete. The displayMilestones function displays the condition of all milestones.

Application Class: This is one of the main classes in our app. It has the following attributes:

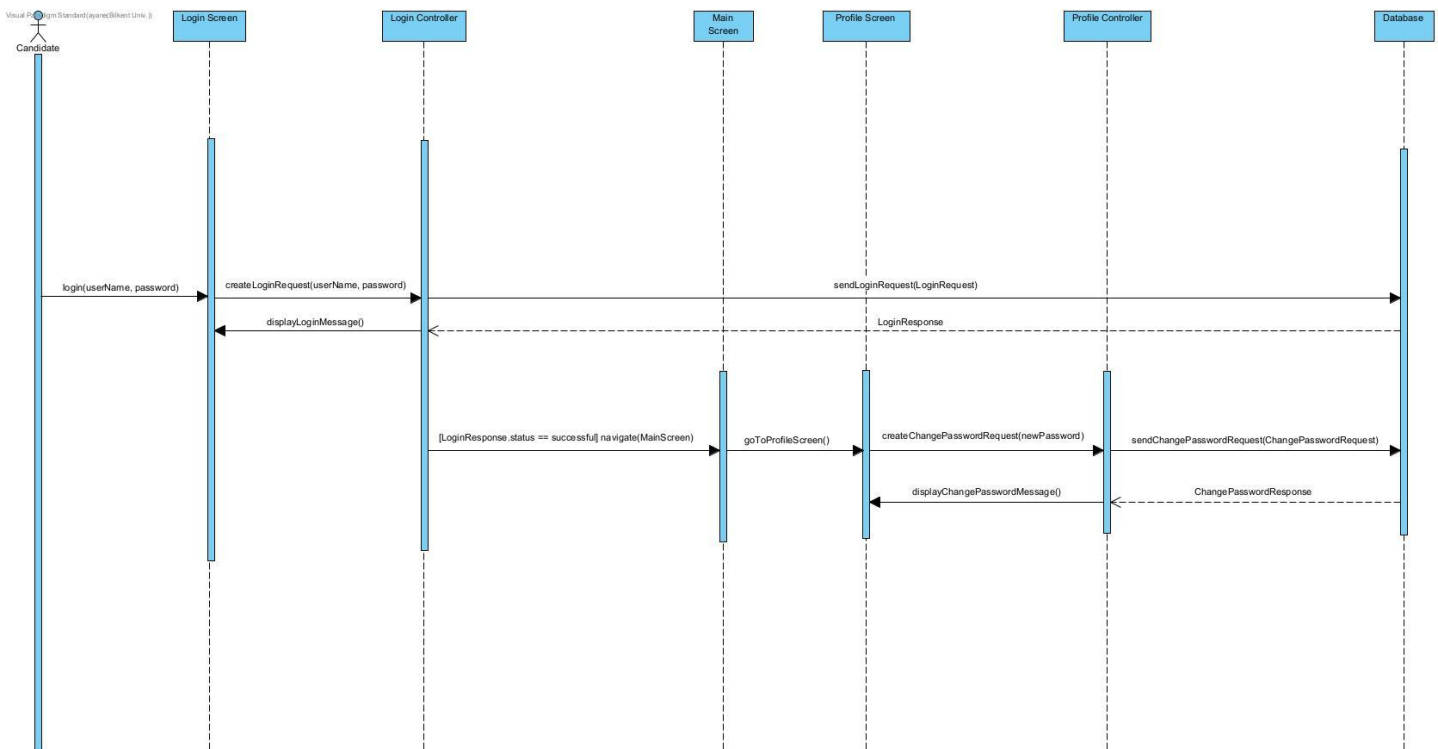
- Candidate User
- DurationPreferred : Represent preferred term for Erasmus
- ApplicationFiles[] : Stores the necessary files for the application process such as statement of purpose, application form and CV
- Semester : The applied semester to the Erasmus program
- Stage : Represents current stage with stageEnum
- IsApproved : Stores the approved condition
- UniversityChoices[] : Stores the university preferences
- ChosenCourses [] : Stores the chosen courses
- ApplicationRoadmap : It is a roadmap for the student process
- IsCVUploaded : Stores the CV status
- IsStatementOfPurposeUploaded : Stores the statement of purpose status
- IsApplicationFilled : Stores the application info filled status
- IsSignedApplicationUploaded : Stores the application upload status

Using this info we can manipulate the application according to user inputs and exportApplicationForm() allows us to fetch the form whenever we want.

TODOList Class: It contains the candidate applications that are assigned to that coordinator. This class is instantiated per coordinator and each coordinator has a list of candidate applications to evaluate and respond. The approveApplication function allows us to approve the applications and the checkCandidateStage(int/string) functions allows us to query the student with the specific ID/name for the coordinator.

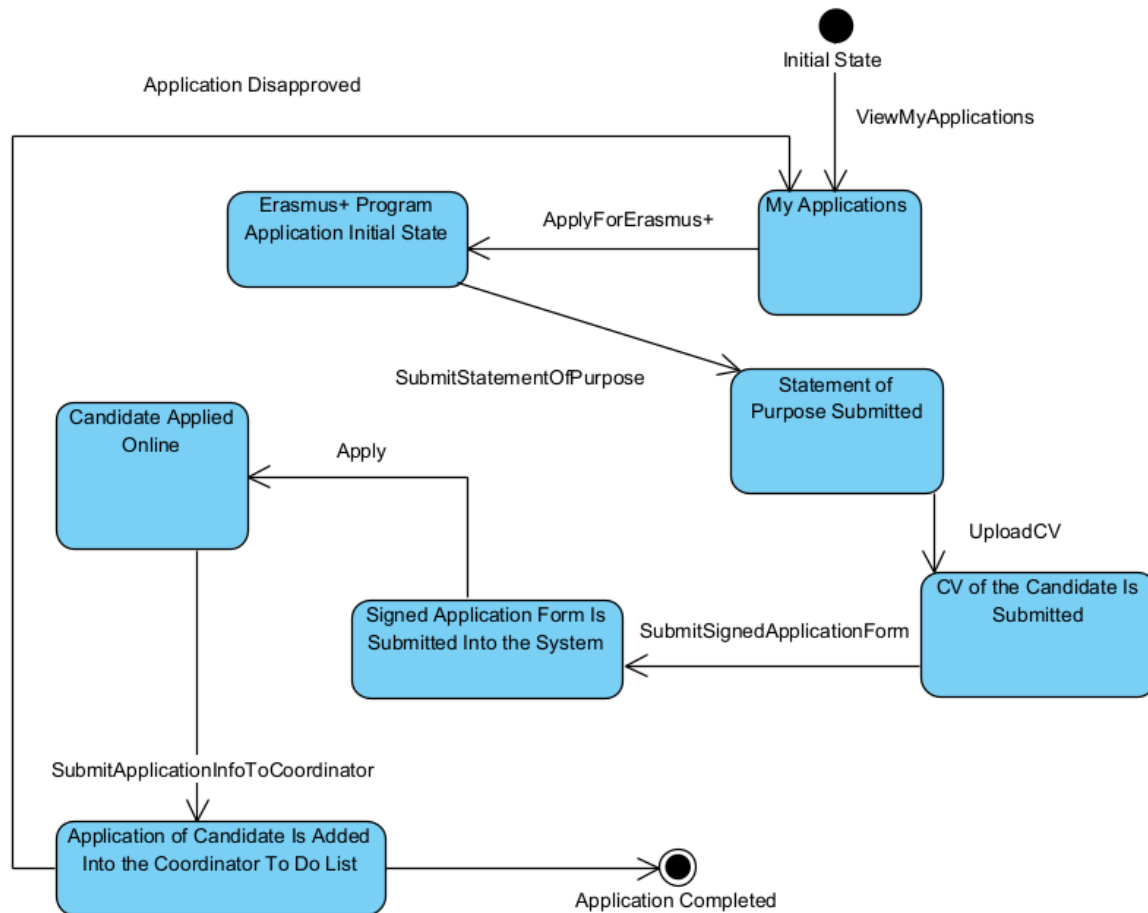
3.3 Sequence Diagram

Below is the sequence diagram for login operation followed by changing password operation. The login screen is the first view that the candidate will be interacting with. By providing a user name and a password to the related fields, the login screen will pass these parameters to the login controller which will create a login request that will be sent to the database and return a result if the provided username at the database matches with the provided password. The login controller will call `displayMessage` function which will display a message based on the login response (login successful / user name - password mismatch / connection error). Additionally, if the login is successful, the login screen will navigate to the main screen where the user can see the profile button with other main interactions. When clicked, the main screen will navigate to the profile screen which is directed by the profile controller class. By providing a new password to the profile screen, the profile controller will create a change password request to the database which will return a change password response. Similar to the login process, a message will be displayed at the screen in order to inform the user. A similar sequence diagram with different view and controller classes can be used to express another profile management related operations.



3.4 State Diagrams

3.4.1 Online Application State Diagram



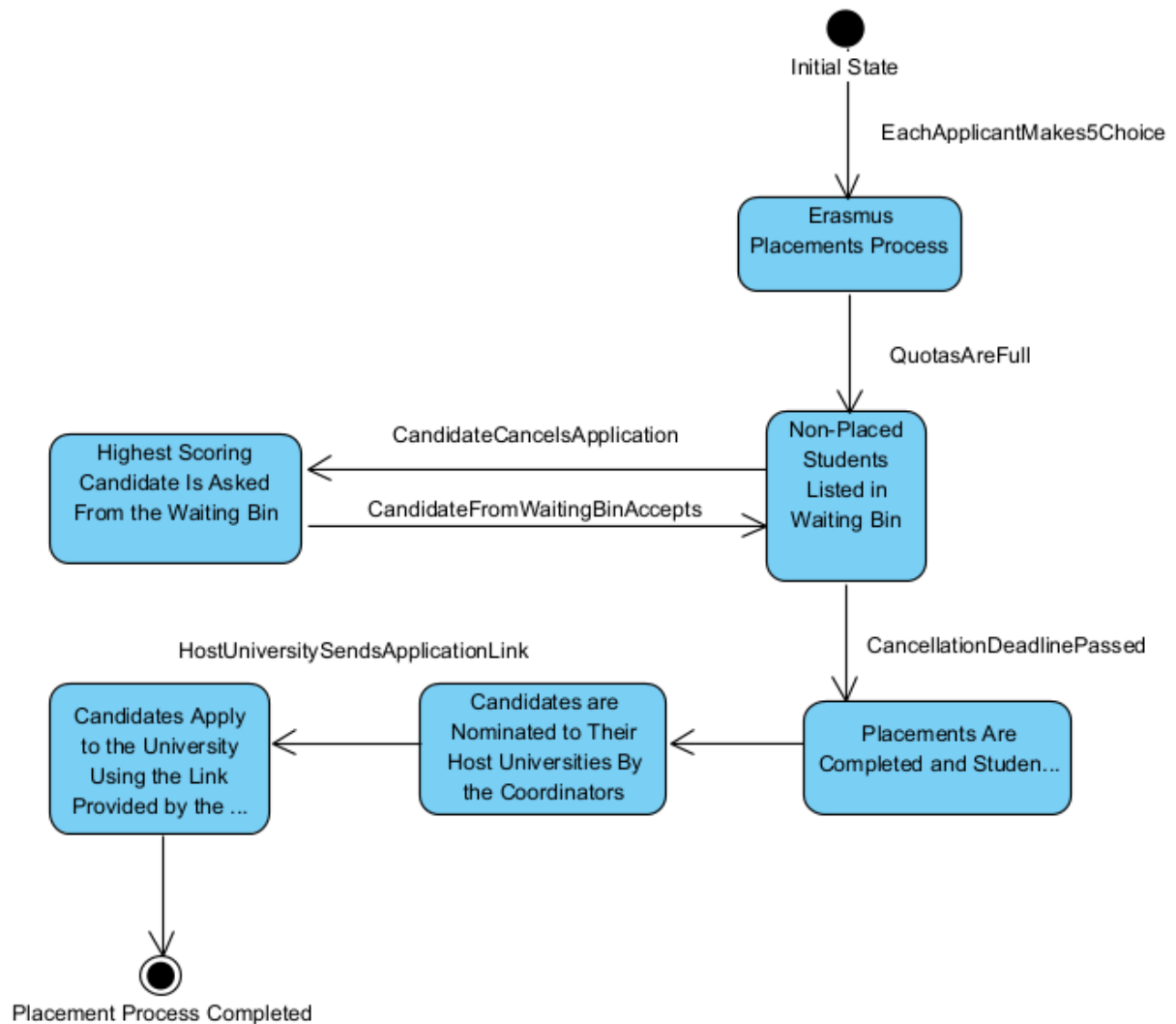
Initial State: A Candidate is to Apply for Erasmus+ Program.

Erasmus+ Program Application Initial State: Assuming that the semester is open for applications, for undergraduates, if a candidate has a minimum CGPA of 2.50 and currently in his/her minimum 3rd or maximum 5th registration semester, and for graduates if he/she is in his/her 1st semester and has a CGPA of 3.00 at the end of the semester, the candidate is eligible to submit their applications online in this state.

Candidate Applied Online: Candidates officially submitted all the required documents into the system, which are namely: statement of purpose, CV of candidates, signed application form.

Application of Candidate is Added into the Coordinator To Do List: The departmental exchange coordinator will either approve or disapprove the application of the candidate.

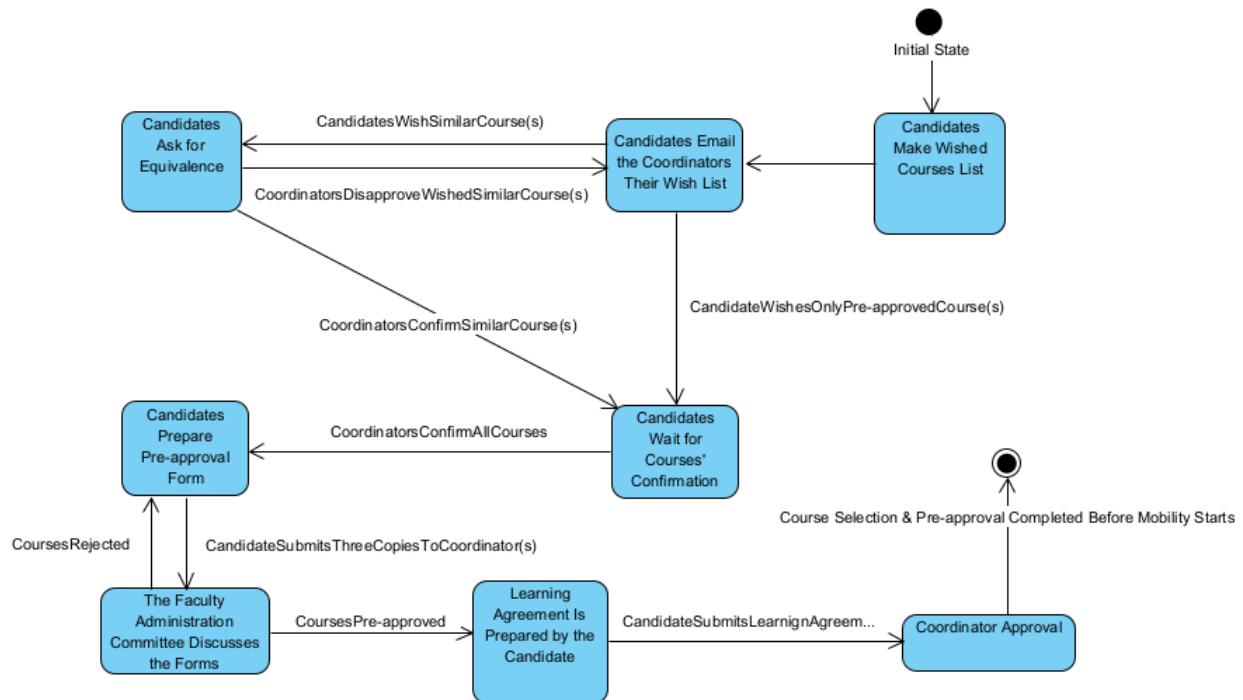
3.4.2 Placements State Diagram



Initial State: A student completed their application to the system and got approved by the departmental exchange coordinator.

Erasmus Placement Process: Top scoring student goes to first choice, quota for that university is decremented.

3.4.3 Course Selection State Diagram



Initial State: Course selection process begins after candidates are placed to host universities.

Candidates Make Wished Courses List: Outgoing students are required to observe previously accepted/rejected courses in their respective assigned host university, and make a list of courses they wish to take. Total credits must be at least 30 ECTS for Erasmus+ Program, and cannot exceed by much.

Candidates Email the Coordinators Their Wish List: The students email the coordinators for their wish list and link to the course web pages (including syllabus). If a candidate wishes courses that are all pre-approved, they go to the state of "Candidates Wait for Courses' Confirmation". If the candidate has any wished for a similar course, they go to "Candidates Ask for Equivalence" state.

Candidates Ask for Equivalence: In this state, coordinators either reject wished similar courses and ask for candidates to find new ones, or approve all similar courses. For most courses, candidates email the Bilkent instructor responsible of the said course and cc's the coordinators, send syllabus and ask for equivalence. For elective courses, candidates need to email coordinators with the syllabus of the said course.

Candidates Wait for Courses' Confirmation: In this state, candidates are to take courses that consist of pre-approved courses from past years, and recently confirmed similar courses (which

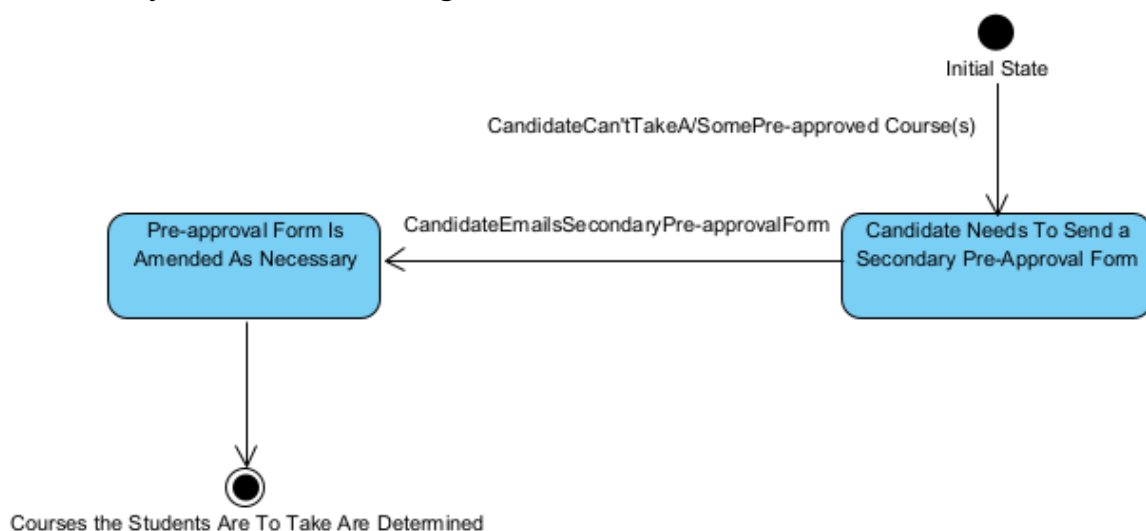
are asked for equivalence). The next state is then comes after candidates have all courses confirmed by the coordinators.

Candidates Prepare Pre-approval Form: Candidates prepare pre-approval form in typed, not handwritten (as such forms will be rejected).

Learning Agreement is Prepared By the Candidate: Learning Agreement includes the list of courses that are in the pre-approval form.

Coordinator Approval: Candidates submit the learning agreement for coordinator approval.

3.4.4 Mobility Process State Diagram

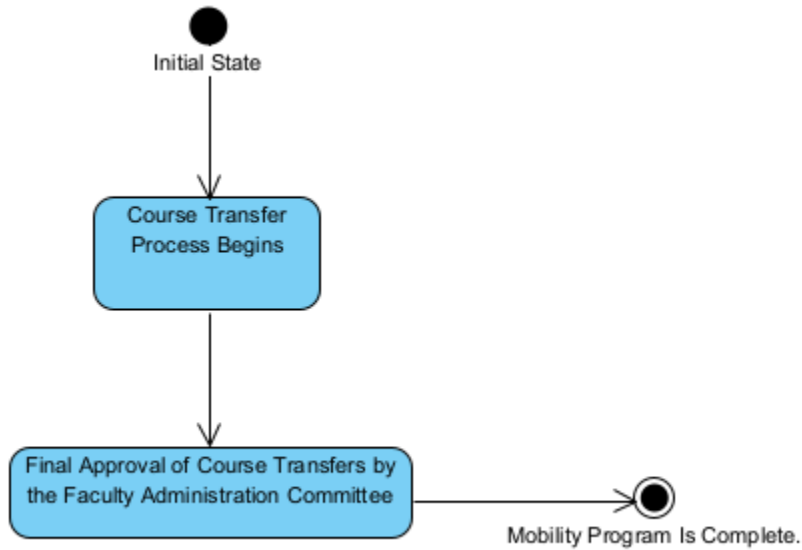


Initial State: Initial state is that mobility already started at the host institution and the pre-approved courses could not be taken due to schedule conflict or prerequisites.

Candidate Can't Take a/some Pre-approved Course(s): The candidates immediately email the coordinators and propose alternative courses.

Courses the Students Are To Take Are Determined: At this state, the courses that will be taken by the students are determined certainly.

3.4.5 Course Transfer State Diagram

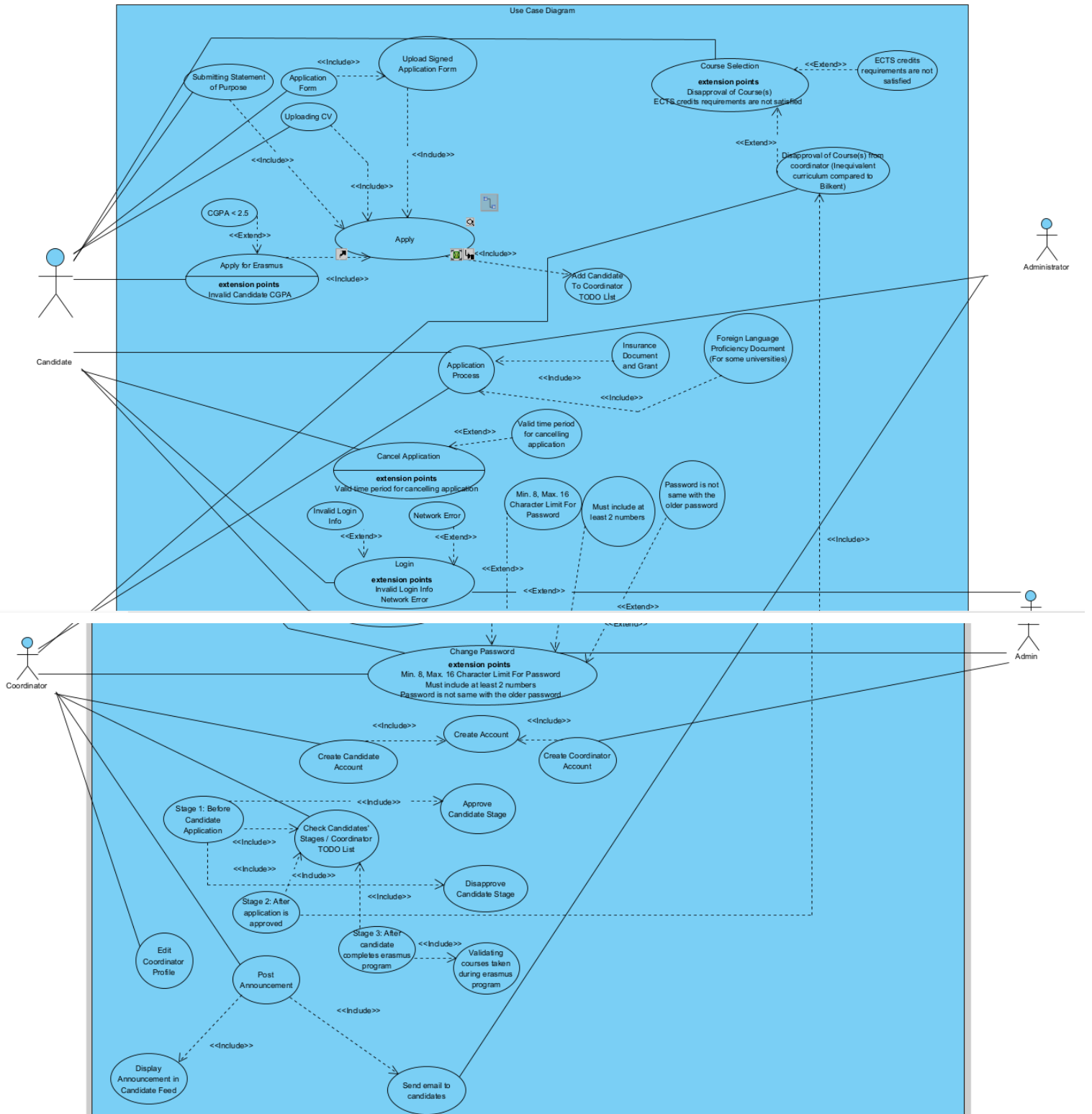


Initial State: At this state, the students have finished their studies.

Course Transfer Process Begins: The official transcript will be sent by the host organization to the Bilkent International Students Office, then forwarded to the coordinators. Then, the coordinators prepare the final course transfer form based on the official transcript and the pre-approval status of the courses.

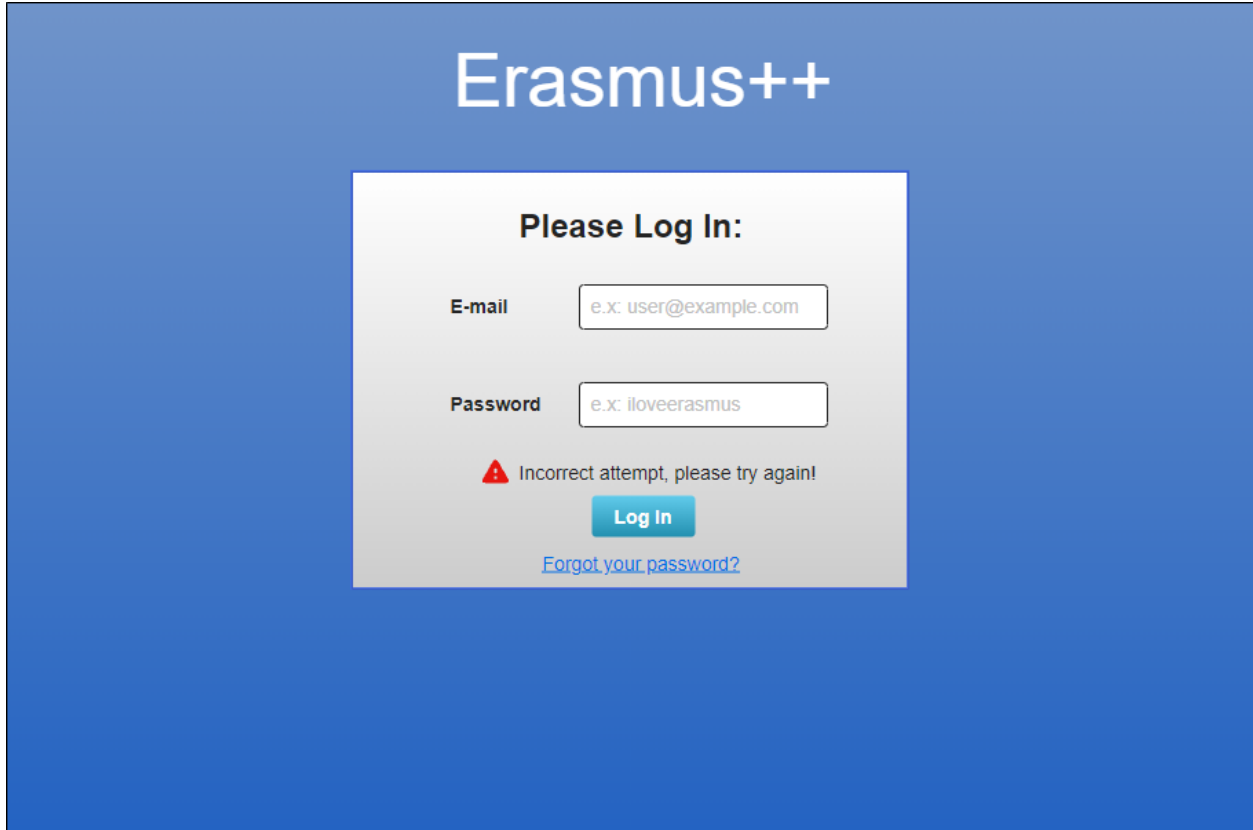
Final Approval of Course Transfers by the Faculty Administration Committee: The coordinators submit the course transfer list to the Faculty Administration Committee for final approval. The Faculty Administration Committee sends the list of transferable courses to the student services.

3.5 Use Case Diagram



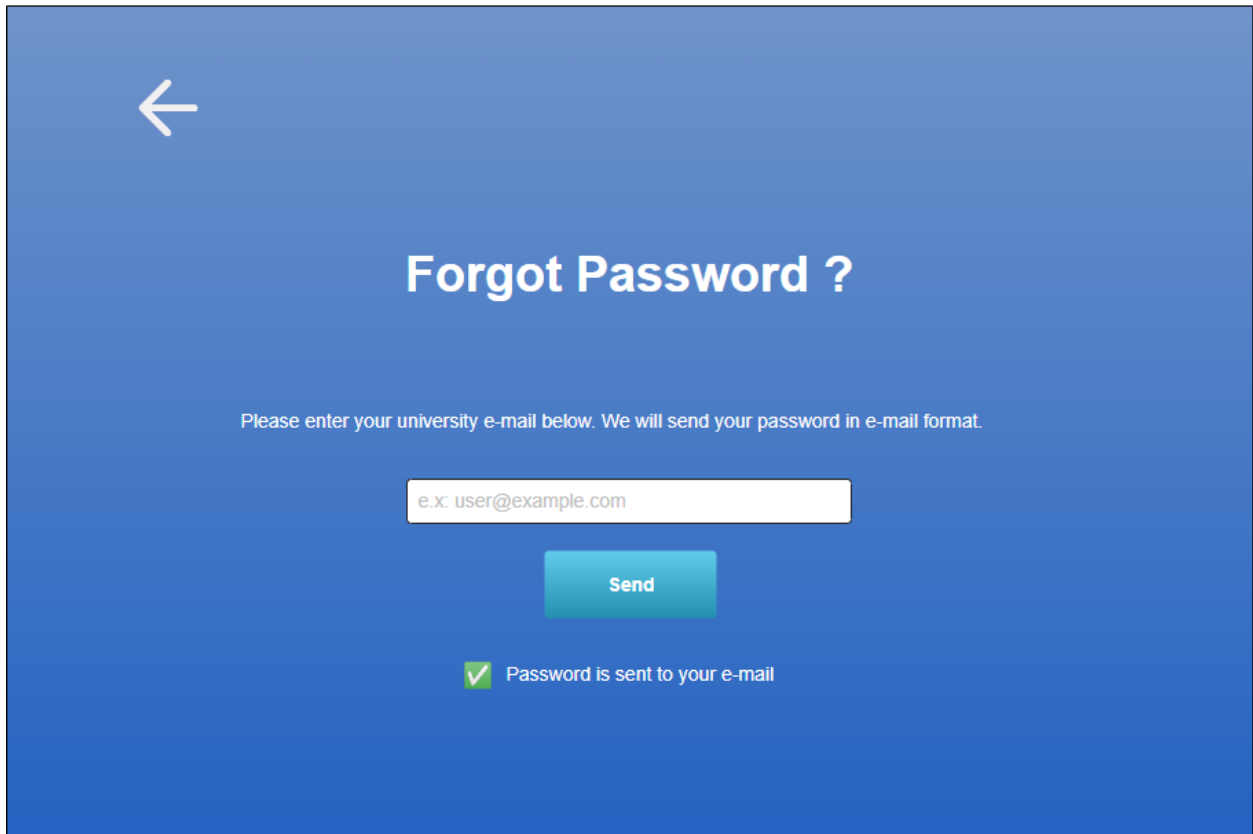
4.0 User Interface

4.1 Login Screen

The image shows a login screen for a system named "Erasmus++". The background is a solid blue color. At the top center, the text "Erasmus++" is displayed in a large, white, sans-serif font. Below this, centered on the screen, is a white rectangular box with a thin blue border. Inside this box, the text "Please Log In:" is centered at the top in a bold, black font. Below the title, there are two input fields. The first is labeled "E-mail" in bold black text, and the second is labeled "Password" in bold black text. Both fields contain placeholder text in a smaller, grey font: "e.x: user@example.com" for the email and "e.x: iloveerasmus" for the password. Below the password field, there is a red warning icon (a triangle with an exclamation mark) followed by the text "Incorrect attempt, please try again!". At the bottom of the white box, there is a blue button with the text "Log In" in white. Below the button, there is a blue hyperlink that reads "Forgot your password?".

This is the login screen. There will be no different login screens for different actor (i.e. Candidate, Coordinator, Admin). An error message will appear in the case of an incorrect log in. If the users forget their password, they can click the “Forgot your password?” which will redirect them to a different screen.

4.2 “Forgot your password?” Screen



←

Forgot Password ?

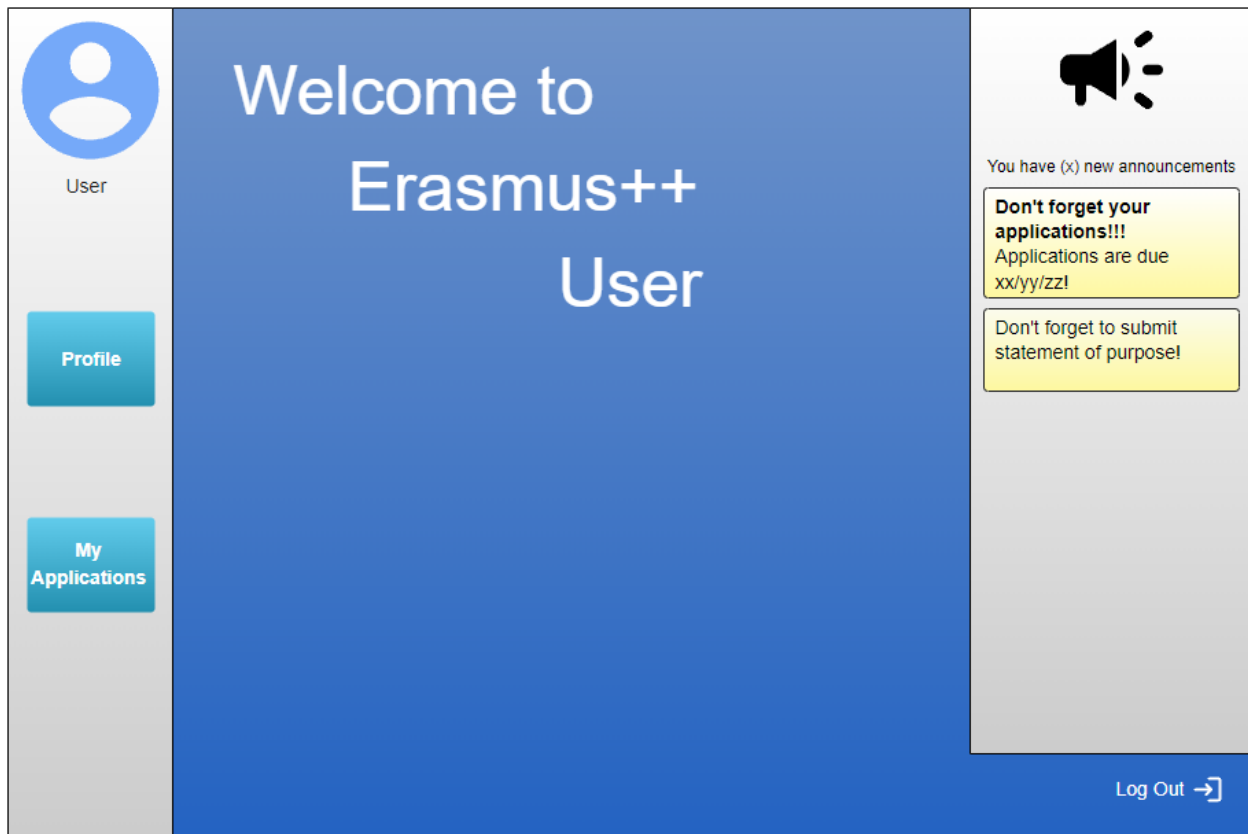
Please enter your university e-mail below. We will send your password in e-mail format.

Send

✓ Password is sent to your e-mail

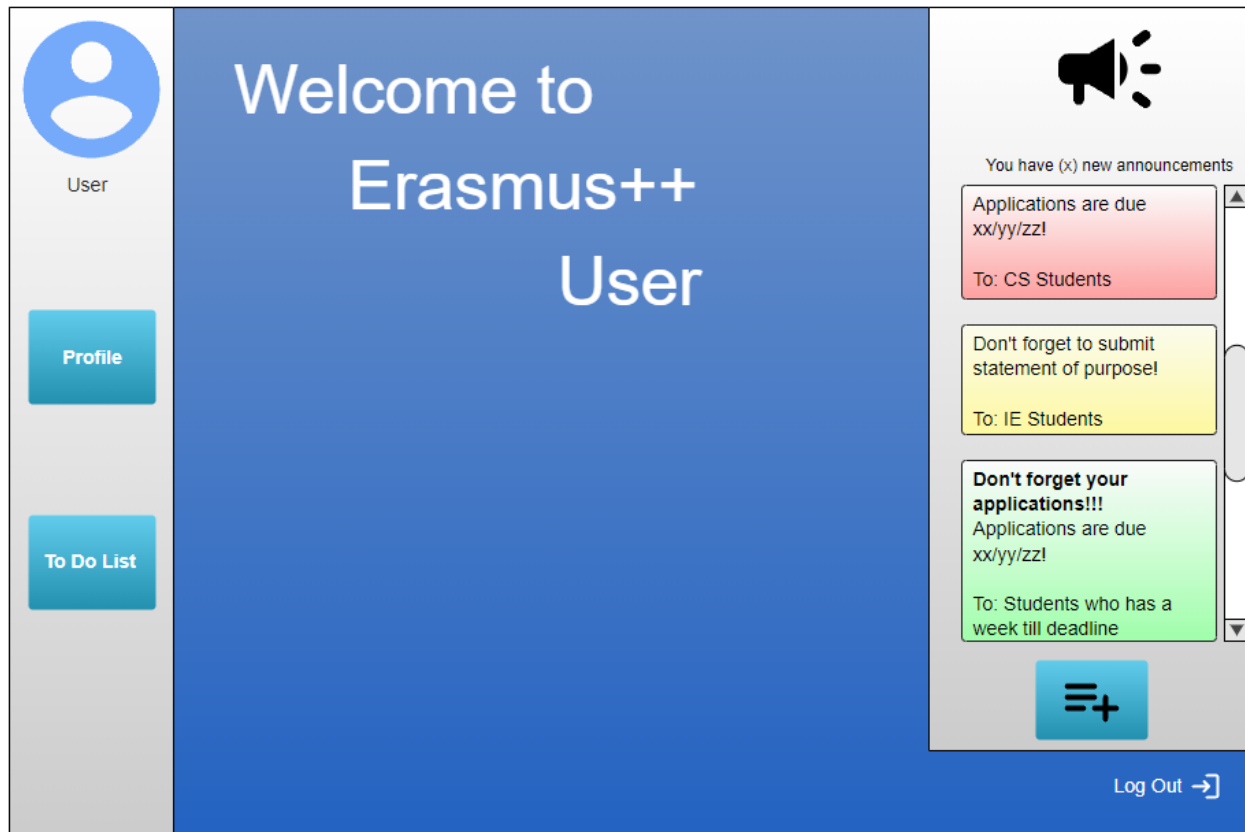
Clicking the “Forgot your password?” will redirect you to this screen. Here, you can acquire a new password by entering your registered e-mail.

4.3 Main Screen (For candidates)



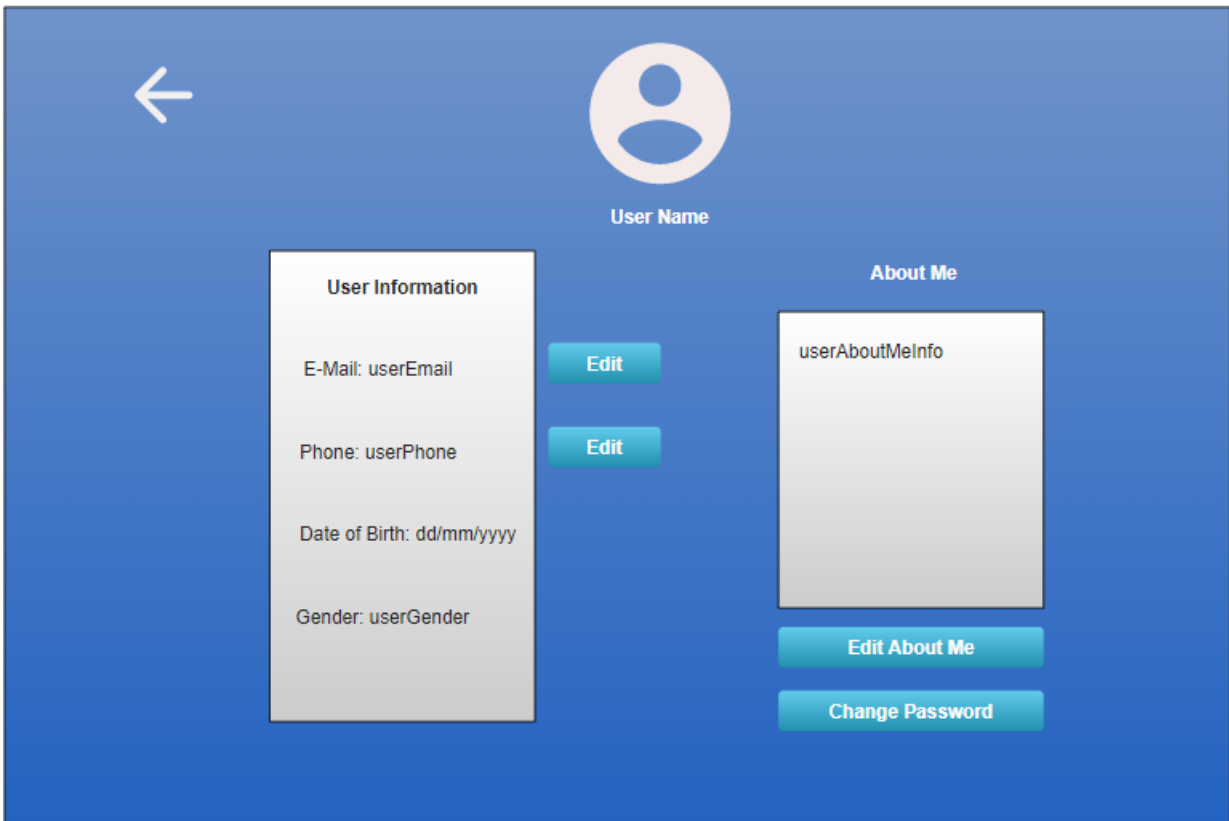
After successfully logging in, candidates will be redirected to this main screen. From here, candidates can logout back to the login screen. Enter their profile, Check out their application process on the left. And, they can view any announcement that they might have received on the right.

4.4 Main Screen (For coordinators)



After successfully logging in, coordinators will be redirected to this main scene. From here, coordinators can go to their Profile and To Do List from the left menu. From the right they can create new announcements for candidates.

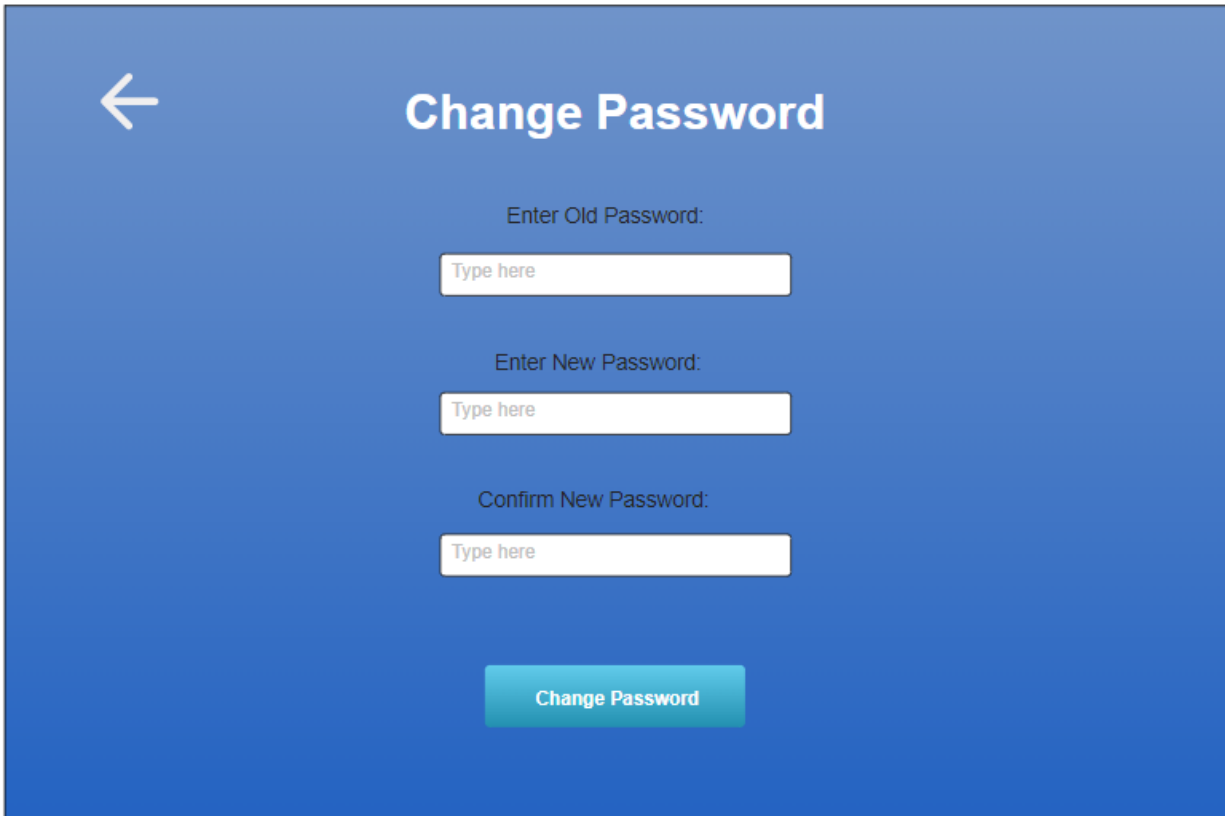
4.5 Profile Screen



The image shows a user profile screen with a blue background. At the top left is a white back arrow. At the top center is a white circular profile icon with the text "User Name" below it. On the left is a white box titled "User Information" containing the following fields: "E-Mail: userEmail", "Phone: userPhone", "Date of Birth: dd/mm/yyyy", and "Gender: userGender". To the right of each field is a teal "Edit" button. On the right is a white box titled "About Me" containing the text "userAboutMeInfo". Below this box are two teal buttons: "Edit About Me" and "Change Password".

Profile screen provides the users to edit their personal information and “About Me” section. Also, they can change their password through this screen.

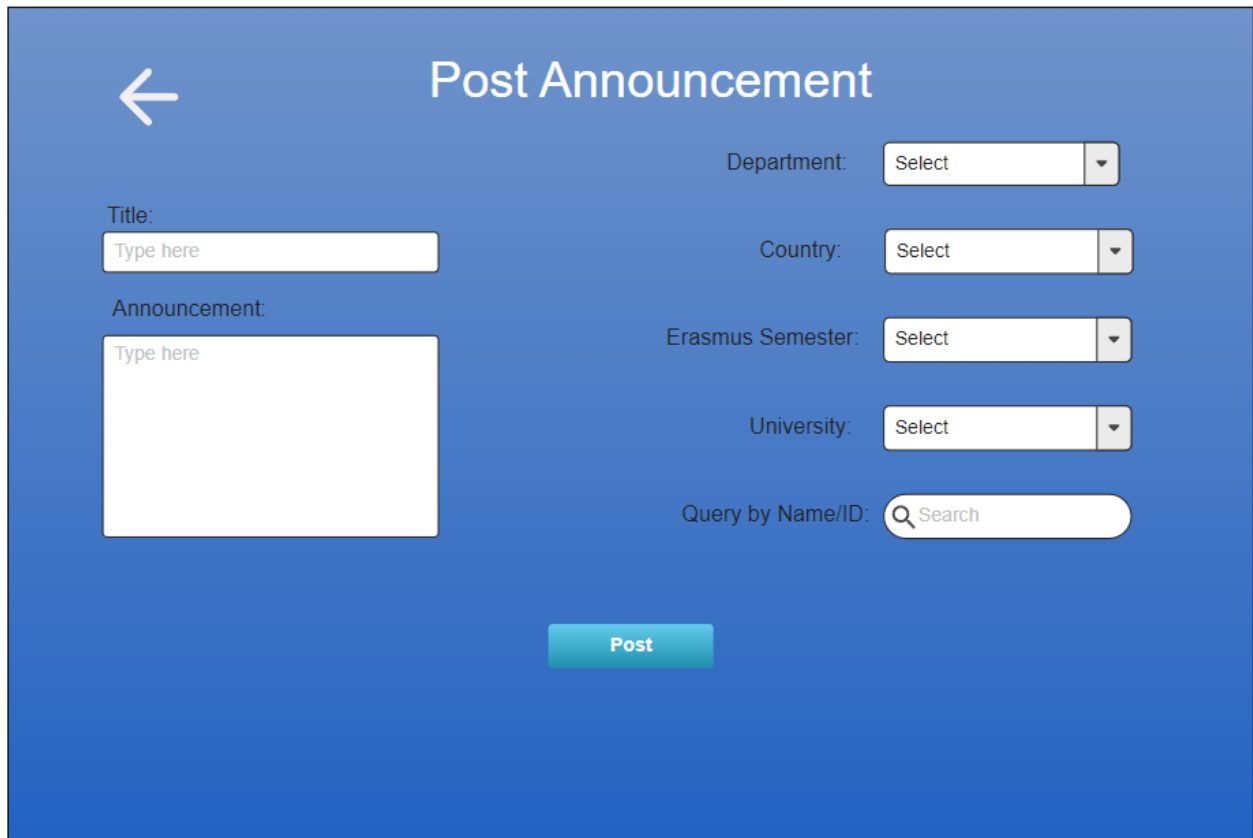
4.6 Change Password Screen



The image shows a 'Change Password' screen with a blue gradient background. In the top left corner, there is a white left-pointing arrow. The title 'Change Password' is centered at the top in a large, bold, white font. Below the title, there are three input fields, each preceded by a label in a smaller white font: 'Enter Old Password:', 'Enter New Password:', and 'Confirm New Password:'. Each input field is a white rectangle with the placeholder text 'Type here' in a light gray font. At the bottom center of the screen, there is a teal-colored button with the text 'Change Password' in white.

Users can change their passwords through this screen.

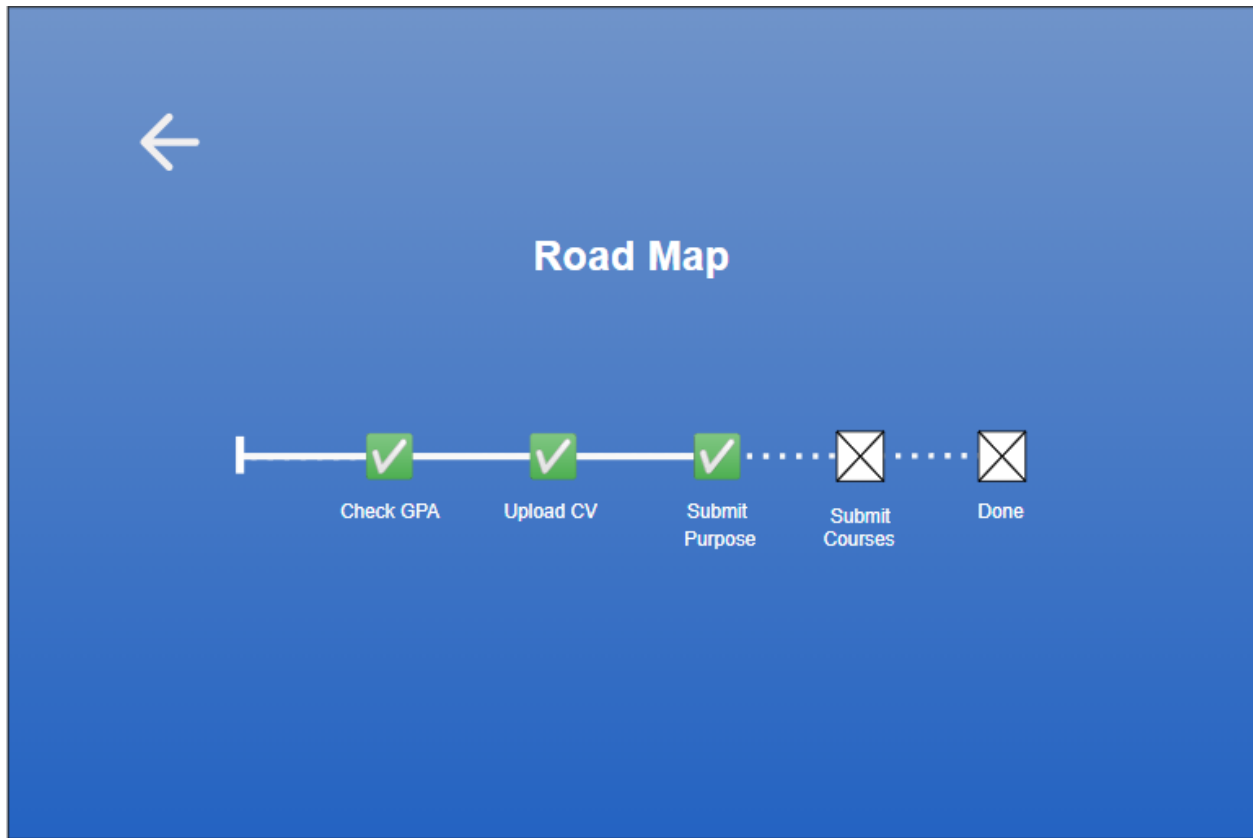
4.7 Post Announcement



The image shows a web interface for posting an announcement. It has a blue background. At the top left is a white back arrow. To its right is the title 'Post Announcement' in white. On the left side, there are two text input fields: 'Title:' with a placeholder 'Type here' and 'Announcement:' with a placeholder 'Type here'. On the right side, there are four dropdown menus: 'Department:', 'Country:', 'Erasmus Semester:', and 'University:', each with a 'Select' option and a downward arrow. Below these is a search bar labeled 'Query by Name/ID:' with a magnifying glass icon and the text 'Search'. At the bottom center is a green button with the text 'Post'.

The coordinators can post announcements using this interface. There are 4 filters and a search by name option to filter down the candidates the announcement will arrive. If the announcement is for everyone no filter should be selected

4.8 Road Map



The candidates can follow their progress in the application process through the road map and see what they have to do next. This screen provides information about the work that must be completed by the candidate. This screen is reached, through the button in 4.10 My Application View.

4.9 To Do List

The screenshot shows a web interface titled "To Do List" with a blue background. On the left, there is a white box labeled "Filtered Applications" containing a scrollable list of three student applications. Each application card displays the student's name, the Erasmus Semester (2023 Spring), and two buttons: "View" and "Approve" (with a red 'X' icon). The third card is highlighted with a blue border and a blue dot. To the right of the list, there are two dropdown menus: "Stage: Select" and "Erasmus Semester: Select".

← To Do List

Filtered Applications

Stage: Select

Erasmus Semester: Select

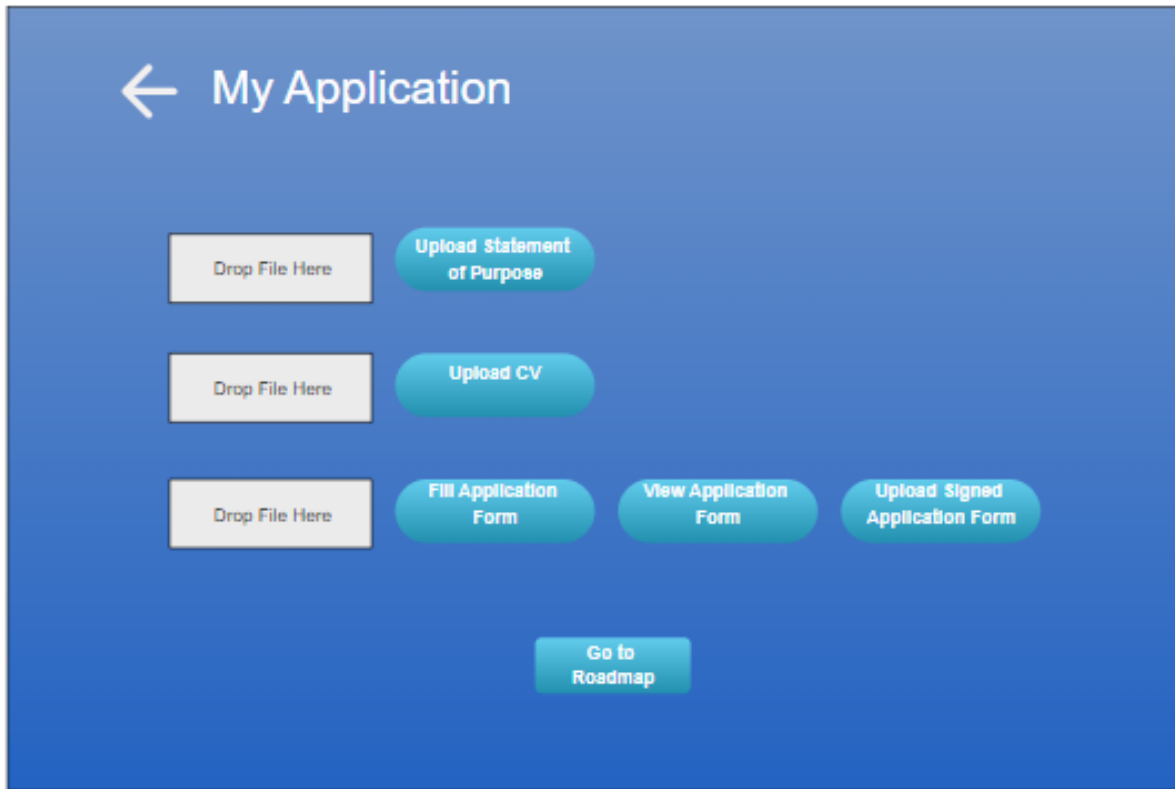
Mehmet Feyyaz Küçük
Erasmus Semester: 2023
Spring
View Approve X

Yunus Türkeri
Erasmus Semester: 2023
Spring
View Approve X

Ender Utlü
Erasmus Semester: 2023
Spring
View Approve X

Coordinators approve using this interface. They can filter the students that are at a higher priority and inspect them accordingly.

4.10 My Application View



The screenshot shows a web interface titled "My Application" with a back arrow icon. It features three rows of file upload areas, each labeled "Drop File Here" in a light gray box. To the right of each "Drop File Here" box is a blue button with white text. The first row has a button labeled "Upload Statement of Purpose". The second row has a button labeled "Upload CV". The third row has three buttons: "Fill Application Form", "View Application Form", and "Upload Signed Application Form". At the bottom center of the interface is a blue button with white text labeled "Go to Roadmap".

The students can upload a statement of purpose, and CV. Also, the students can click “Fill Application Form”, view that filled form, and upload the signed version of that form. The roadmap screen view is available through the “Go to Roadmap” button.

4.11 Application Form Screen (Edit)

The screenshot shows a web application titled "Application Form" with a back arrow in the top left. The form is divided into several sections:

- Personal Information:**
 - Firstname: Mehmet Feyyaz
 - Lastname: KÜÇÜK
 - T.C. ID: 12345678987
 - Date of Birth: 28.02.2002
 - Nationality:
- Student Information:**
 - Student Number: 22003550
 - Department/Class: Computer Science/2
 - Degree: Bachelor's
 - Cumulative GPA: 3.31
- Contact Information:**
 - University E-mail: feyyaz.kucuk@bilkent.edu.tr
 - Personal E-mail:
 - Phone - Mobile: 05358714993
 - Phone - Local:
 - Postal Address:
- Erasmus Preferences:**
 - University Preferences: 1: , 2: , 3: , 4: , 5:
 - Duration Preferred: ☐ Fall ☒ Spring

A blue "Submit" button is located at the bottom center of the form.

In this screen, candidates can fill out the application form needed for applying. Some of the information will be available already, so those parts will not be editable. Candidates need to fill out the information that the system does not have and their erasmus preferences.

4.12 Application Form Screen (View)

Application Form (View Mode)

Personal Information:
Firstname: Ender
Lastname: UTLU
T.C. ID: 98723409812
Date of Birth: 28.02.2002
Nationality: Turkey

Student Information:
Student Number: 22002475
Department/Class: Computer Science/2
Degree: Bachelor's
Cumulative GPA: 3.69

Contact Information:
University E-mail: ender.utlu@bilkent.edu.tr
Personal E-mail: enderutlu1907@gmail.com
Phone - Mobile: 05356662763
Phone - Local: 03122875846
Postal Address: Mustafa Kemal Mahallesi
2019. Sok.
No.14


Erasmus Preferences:
University Preferences:
1: Berlin University
2: Dortmund TU
3: Vrije University
4: Saarland University
5: Ecole Polytechnique University

Duration Preferred: Spring

[Export File](#)

Coordinators and candidates can view this screen however they cannot edit it. Candidates can export this file as a pdf so that they can sign and upload it.

4.13 Course Registration Screen



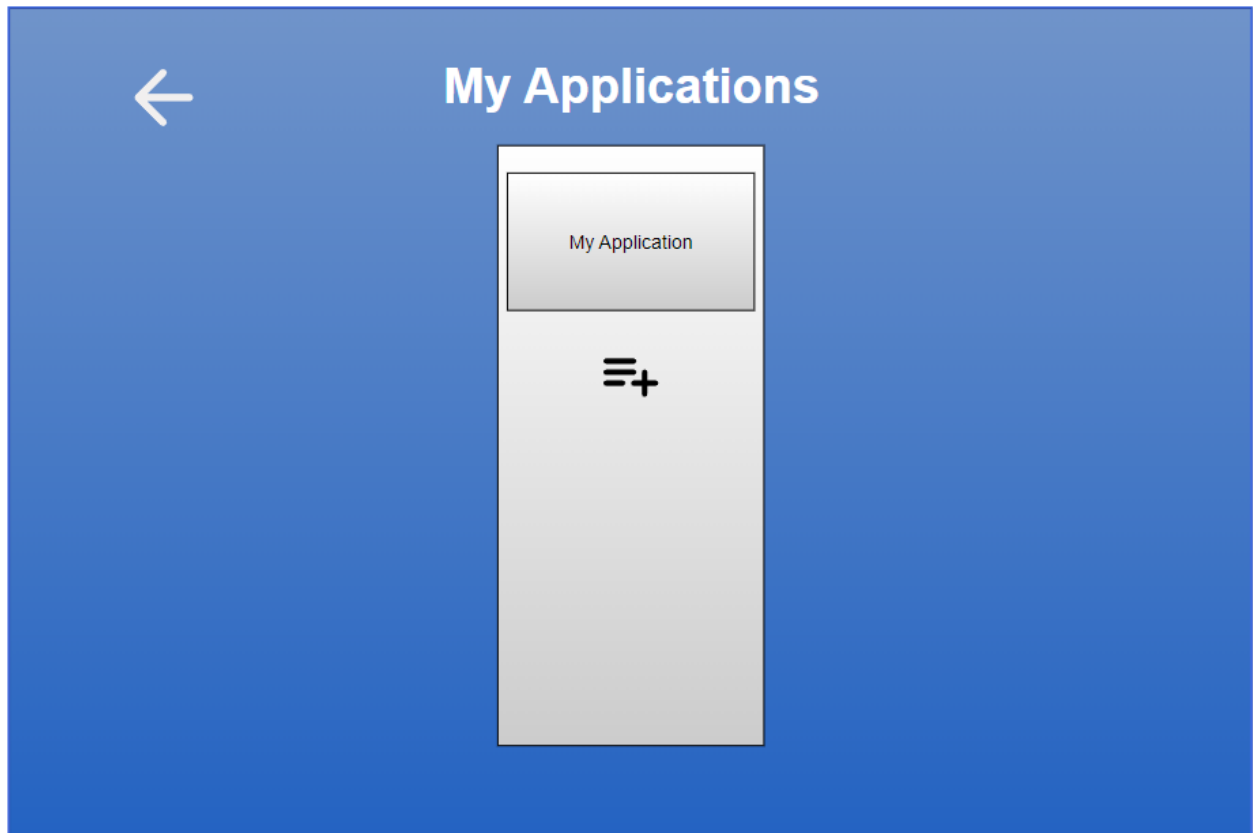
The image shows a 'Course Registration' screen with a blue background. At the top left is a white back arrow. The title 'Course Registration' is centered in white. Below it, 'Total ECTS: 11.5' is displayed. The screen is divided into two columns: 'Bilkent Courses' and 'Equivalent Courses'. Under 'Bilkent Courses', there are two dropdown menus: 'CS342 - Operating Systems' (with a dropdown arrow) and 'CS353 - Database Systems' (with a dropdown arrow). To the right of these are the ECTS values 'ECTS: 5.0' and 'ECTS: 6.5' respectively. Under 'Equivalent Courses', there are two dropdown menus: 'COMP55 - Operating Systems' (with a dropdown arrow) and 'COMP72 - Database Systems' (with a dropdown arrow). At the bottom center is a white plus sign inside a circle. In the bottom right corner is a green 'Submit' button.

Bilkent Courses		Equivalent Courses	
CS342 - Operating Systems	ECTS: 5.0	COMP55 - Operating Systems	
CS353 - Database Systems	ECTS: 6.5	COMP72 - Database Systems	

Submit

In the course registration screen, the candidates can select and add Bilkent courses along with equivalent courses. They can also see the ECTS of the courses and follow the total ECTS through that screen.

4.14 My Applications



This interface will be used to create an application. It will be used by the candidates. Candidates will be able to see their applications and add more if they are allowed to.