Bilkent University

Department of Computer Engineering

CS319 Term Project

Erasmust: Bilkent's Erasmus and Exchange Application

Final Report

Group "BubbleZort" - 2C

Nisa Yılmaz - 22001849

Tolga Özgün - 22003850

Eylül Badem - 22003079

Yahya Eren Demirel - 21903026

Emirhan Büyükkonuklu - 22003049

Barış Yıldırım - 22003175

Instructor: Eray Tüzün

Teaching Assistants: Emre Sülün, Muhammad Umair Ahmed, İdil Hanhan, Mert Kara

Table Of Contents

1. Introduction

In this project, we aimed to develop a web application to facilitate the Erasmus application process for students in our school. Using Spring, React, and PostgreSQL, we have built a user-friendly platform that streamlines the application process and provides relevant information to students.

Throughout the implementation of this project, we accomplished several key objectives. Firstly, we designed a functional and intuitive user interface using React, allowing students to easily navigate the application and find the information they need. Secondly, we implemented a secure and reliable backend using Spring and PostgreSQL, ensuring the confidentiality of student data and the smooth functioning of the application. Thirdly, we integrated various features and tools, such as a document upload system and a notification system, to enhance the user experience and improve the efficiency of the application.

As BubbleZort, we have successfully implemented the Erasmus side of the program. Our program mainly targets a user base which includes Erasmus coordinators, outgoing Erasmus students, Dean, Faculty Administration Board Members and Course Coordinators.

Erasmus coordinators can run their operations related to the Erasmus application period of students on our app. They can open Erasmus periods, add new Partner Schools, adjust the quota of these Partner Schools, review and evaluate Erasmus applications from outgoing students. After that, the system can automatically create a placement list for students and assign them to universities according to their ranks and university requests. System then creates a waiting list for students who did not get assigned to any universities and assigns them to universities with remaining quotas, if any, according to their ranks. Outgoing students can send applications within applicaton deadlines, follow the processes of the applications, view their application via forms and cancel their application. Dean can use the application to view the Course Transfer Forms and sign the forms with a pdf file of their signatures. Faculty Administration Board Members can view applications, track application processes for each student and may take over the work of Erasmus coordinators from time to time. Course Coordinators can review Course Review forms and evaluate it according to the syllabus of the relevant course, they can also leave comments regarding what they see missing and want to add.

Overall, our web application has the potential to significantly improve the Erasmus application process for students in our school, making it more convenient and accessible for them to pursue international opportunities facilitating the process of coordinators and everyone considered as staff. In the following sections of this report, we will delve into the details of our project and discuss the challenges and solutions we encountered along the way.

2. Lessons Learned

In the process of this project we learned a lot about importance of team work and communication as a group. Everyone tried to contribute in the best way they can and was devoted throughout the project. Starting from day one, we had frequent group meetings and every member attended when possible. We allocated work evenly and when a group member was busy, we reallocated work accordingly. No doubt that this project has improved our soft skills such as time management, collaboration, decision making and also we cannot ignore the improvement in our solution domain knowledge we gained while implementing the code. This project truly contributed a lot and we will see these benefits both in our careers and social lives.

To elaborate on hard skills, we expanded our knowledge in the languages we used. We learned how to create a project from scratch. And in each report we learned how to generate UML diagrams, and became more involved with our software engineer side rather than our programmer side, which was the real aim of the course.

Although we learned a lot, this does not imply we could not have done things differently and we cannot have made this project experience to go more smoothly. As a group we could have started to implement the project sooner. This would have reduced the tension between group members and evidently increased our efficiency. As we came closer to the demo date, our workload seemed more overwhelming and we doubted that we would present a great project. But as said, these feelings would have been prevented by just starting earlier and scheduling meetings more frequently.

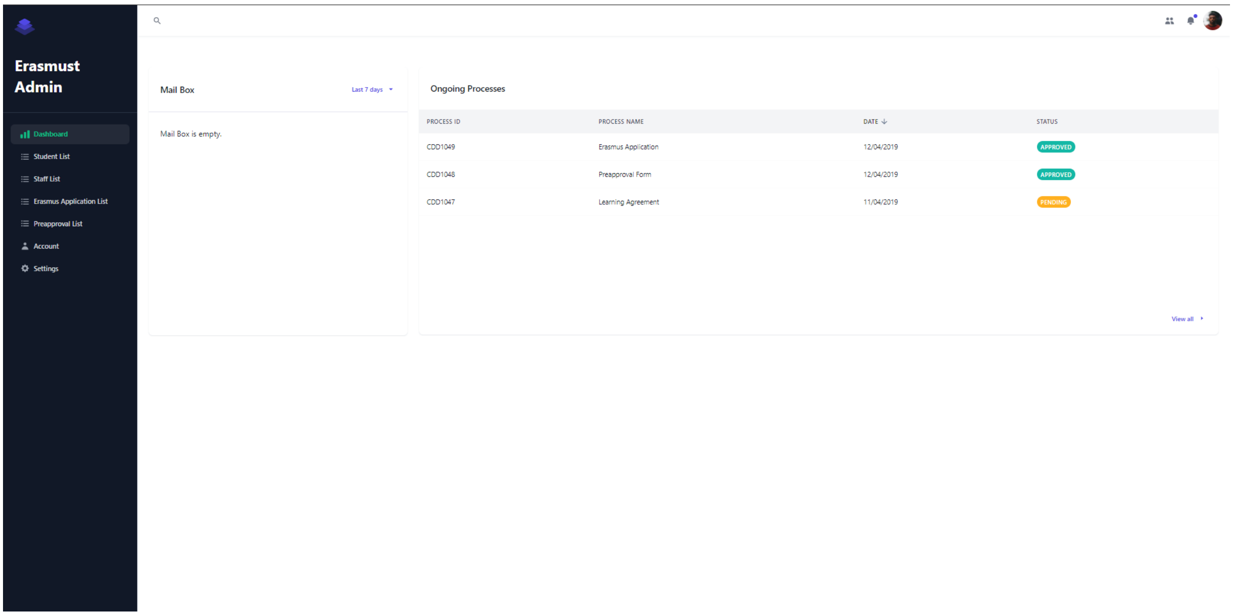
If we are to work together again as as group in the feature, we will keep in mind the things we said we could have done differently, and we believe that we will have a much more systematic and less stressful work process since we know each other better and are familiar with the way these six people work, their characters and roles in a group.

3. User's Guide

3.1. Admin

Admins can create Students and Staff members like Erasmus Coordinator, Faculty Administration Board Member, Dean and Course Coordinator. They can view the list of these users at their respective CRUD pages to which they can route from the sidebar. Admins can also view all of the forms submitted to the system and their respective attributes.

In order to access the app, Admins are required to log in with their mail and their password. By doing so they are directed to their respective dashboard page which contains recent information about ongoing processes and their status. They can access pages like Student List, Staff List; Erasmus Applications List, Exchange Applications List, Preapproval List, Learning Agreement List, Course Transfer List. They can perform CRUD operations in both Student and Staff List but in Forms' respective pages they can only view the list of forms created in the system. They can safely log out from their accounts like all other users do.



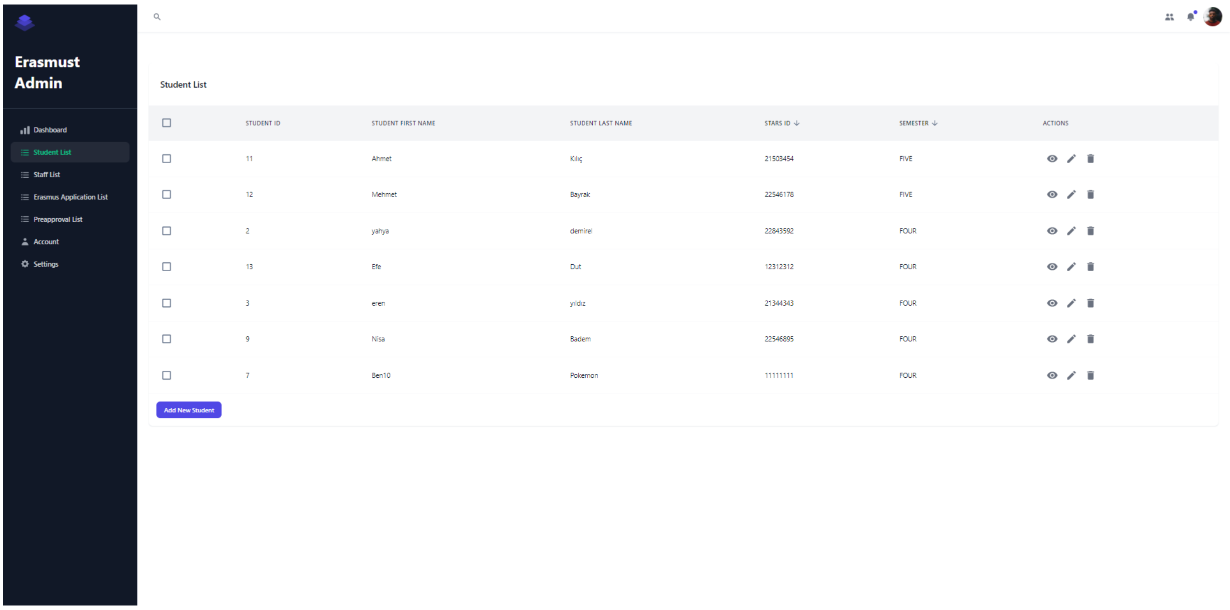
Mail Box

Mail Box is empty.

Ongoing Processes

PROCESS ID	PROCESS NAME	DATE ↓	STATUS
CDD1049	Erasmus Application	12/04/2019	APPROVED
CDD1048	Preapproval Form	12/04/2019	APPROVED
CDD1047	Learning Agreement	11/04/2019	PENDING

Fig. 1: Admin Dashboard

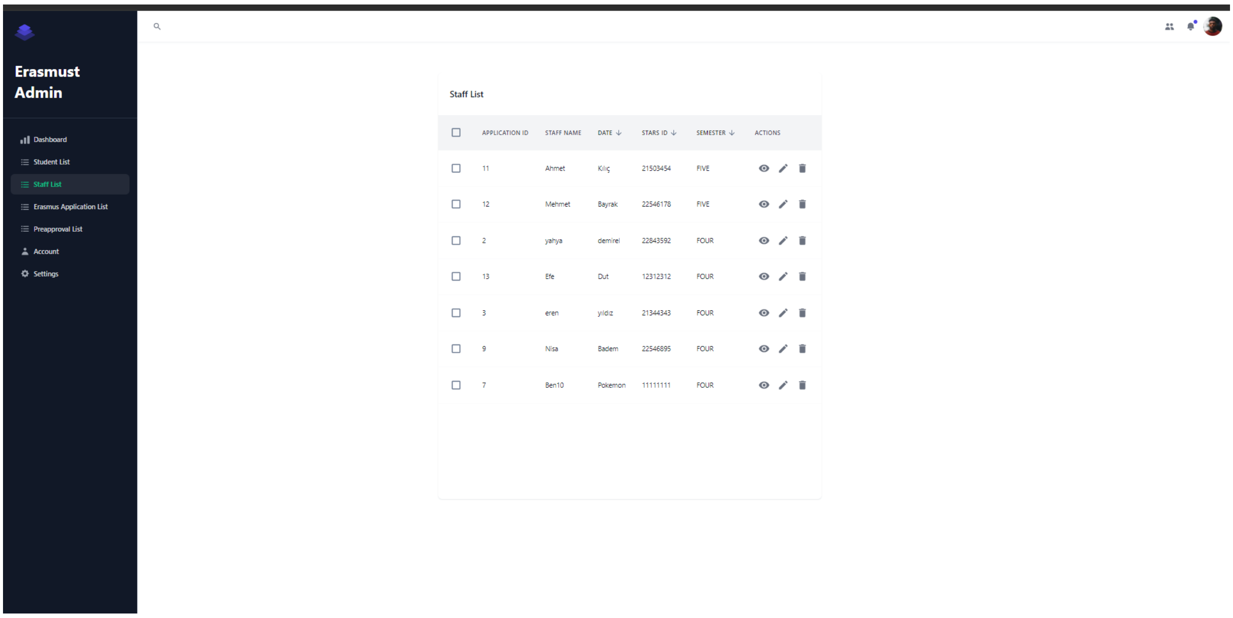


Student List

□	STUDENT ID	STUDENT FIRST NAME	STUDENT LAST NAME	STARS ID ↓	SEMESTER ↓	ACTIONS
□	11	Ahmet	Kılıç	21503454	FIVE	
□	12	Mehmet	Bayrak	22546178	FIVE	
□	2	yahya	demirel	22843592	FOUR	
□	13	Efe	Dut	12312312	FOUR	
□	3	eren	yıldız	21344343	FOUR	
□	9	Nisa	Badem	22546895	FOUR	
□	7	Ben10	Pokemon	11111111	FOUR	

Fig. 2: Admin Student List

Admins can view all of the Students in the system. They can view, edit or remove Students using this page.

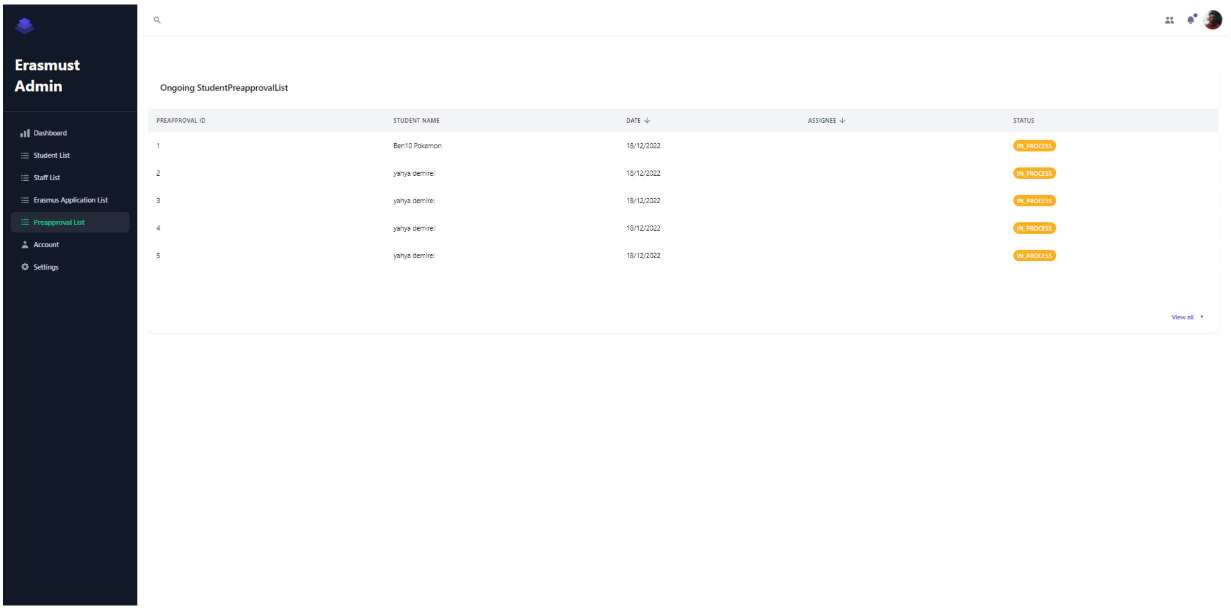


Staff List

☐	APPLICATION ID	STAFF NAME	DATE ↓	STARS ID ↓	SEMESTER ↓	ACTIONS
☐	11	Ahmet	Kılıç	21503454	FIVE	
☐	12	Mehmet	Bayrak	22546178	FIVE	
☐	2	yahya	demirel	22843592	FOUR	
☐	13	Efe	Dut	12312312	FOUR	
☐	3	eren	yıldız	21344343	FOUR	
☐	9	Nisa	Badem	22546895	FOUR	
☐	7	Ben10	Pokemon	11111111	FOUR	

Fig. 3: Admin Staff List

Admins can view all of the Staff in the system. They can edit or remove Staff using this page.

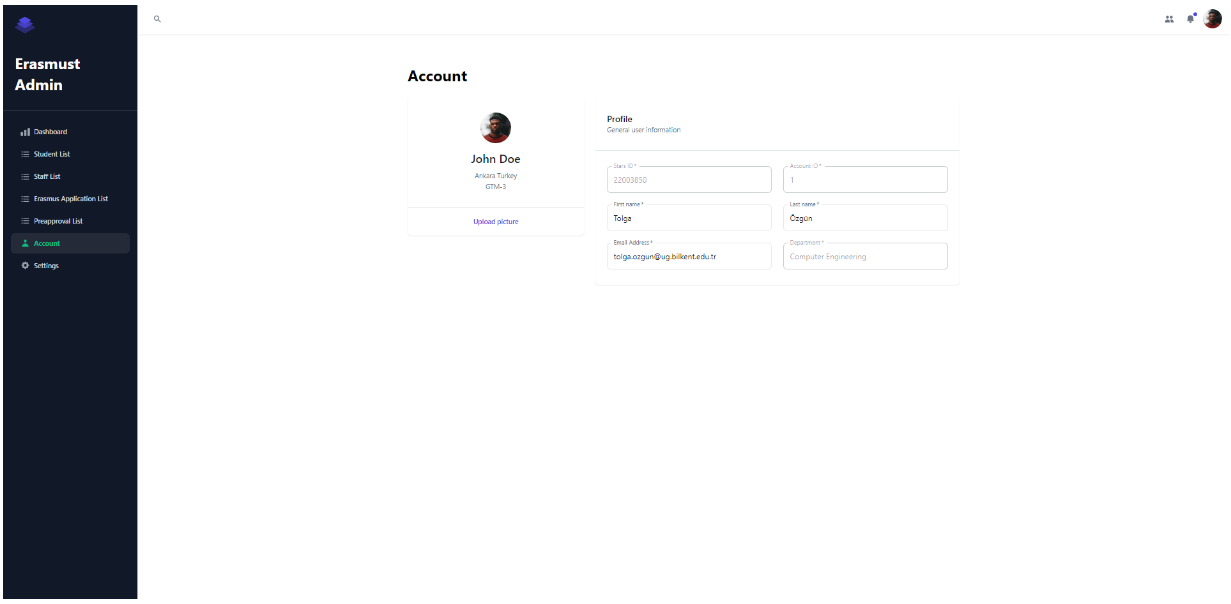


Ongoing StudentPreapprovalList

PREAPPROVAL ID	STUDENT NAME	DATE ↓	ASSIGNEE ↓	STATUS
1	Ben10 Pokemon	18/12/2022		IN_PROCESS
2	yahya demirel	18/12/2022		IN_PROCESS
3	yahya demirel	18/12/2022		IN_PROCESS
4	yahya demirel	18/12/2022		IN_PROCESS
5	yahya demirel	18/12/2022		IN_PROCESS

Fig. 4: Admin Preapproval List

Admins can see the Preapproval submissions to the system. They can review the preapproval forms here.



Account

John Doe
Ankara Turkey
GTM-3

Profile
General user information

Stars ID *: 22003850
Account ID *: 1
First name *: Tolga
Last name *: Özgün
Email Address *: tolga.ozgun@ug.bilkent.edu.tr
Department *: Computer Engineering

Fig. 5: Admin Account

3.2. Staff

Staff is subcategorized into 5 different members. These are Course Coordinator, Erasmus Coordinator, Dean and Faculty Administration Board Member. They have different roles and responsibilities to the system but all of the submembers of Staff are allowed to access certain common pages which include; Dashboard, Settings, Account and School List. In their respective Account page, Staff members can upload a png file of their signatures to sign documents. At School List Page, they can view partner schools.. And finally at Settings page, they can change their passwords.

3.2.1. Course Coordinator

Course Coordinators' work scope is limited to reviewing Course Review Forms so their access to the system is restricted compared to other members. They can view, approve/disapprove a certain Course Review Form sent by an Outgoing Erasmus Student and leave comments regarding what they see missing and want to add. In order for this to be possible, they have access to the Course Review Form and View Course Review Form Page.

3.2.2. Erasmus Coordinator

Erasmus Coordinator has major responsibilities to the system so they have access to various pages. They can create Course Transfer forms, review and sign Erasmus Applications, Preapprovals, Learning Agreements. Then can generate PDF of applications. They can place students into universities, and if there are any universities left, they can do the placement again.

3.2.3. Dean

Dean has less responsibilities to the system compared to other staff members. So their access to system is also very limited. They are required to sign or refuse to sign certain Course Transfer Forms. So they have only access to Course Transfer Form Page.

3.2.4. Faculty Administration Board Member

Faculty Administration Board Member can create Course Transfer forms, and sign them.And can help Erasmus Coordinator about various tasks.

3.3. Outgoing Erasmus Student

Outgoing Erasmus Student can create Erasmus Application and can view all of the previously submitted Erasmus Applications. These are also available for Preapproval Application. They have an Account page in which they can change their mail and a Settings Page in which they can update their password.

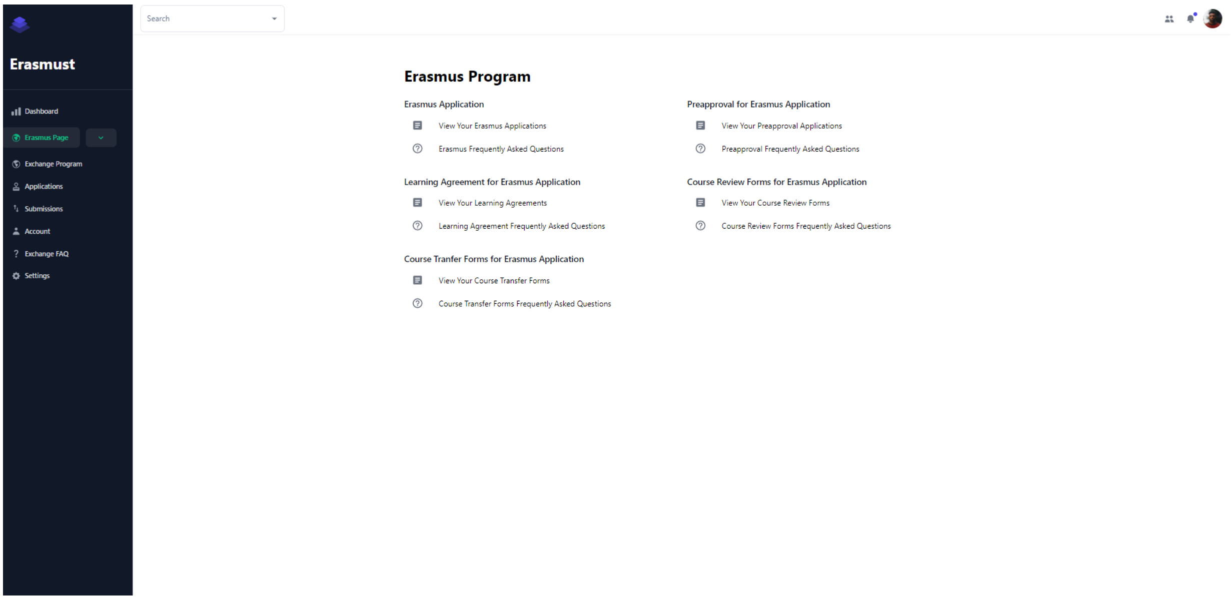
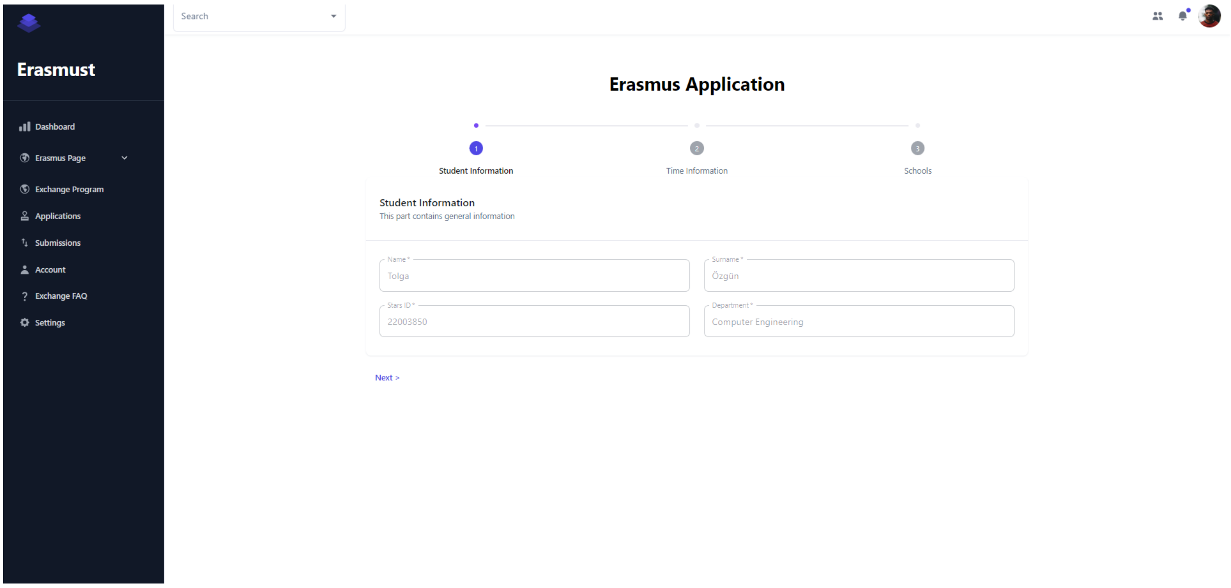

Erasmust
Search
Dashboard
Erasmus Page
Exchange Program
Applications
Submissions
Account
Exchange FAQ
Settings
Erasmus Program
Erasmus Application
View Your Erasmus Applications
Erasmus Frequently Asked Questions
Preapproval for Erasmus Application
View Your Preapproval Applications
Preapproval Frequently Asked Questions
Learning Agreement for Erasmus Application
View Your Learning Agreements
Learning Agreement Frequently Asked Questions
Course Review Forms for Erasmus Application
View Your Course Review Forms
Course Review Forms Frequently Asked Questions
Course Tranfer Forms for Erasmus Application
View Your Course Transfer Forms
Course Transfer Forms Frequently Asked Questions

Fig. 6: Student Erasmus Program



Erasmus Application

Student Information
This part contains general information

Name * Tolga

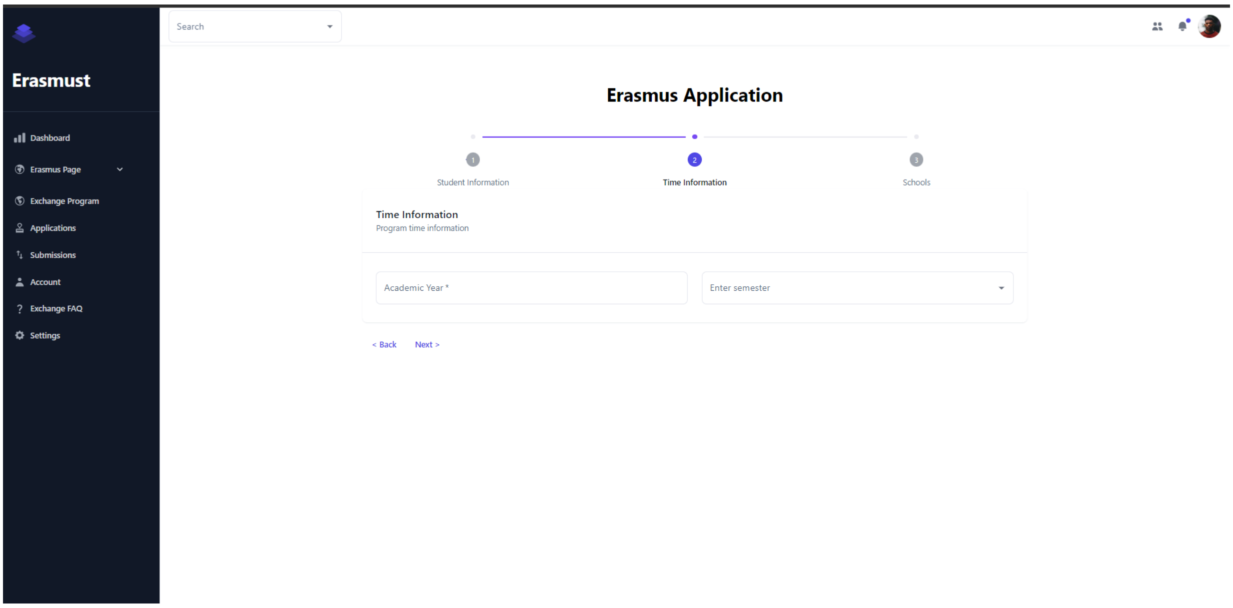
Surname * Özgün

Stars ID * 22003850

Department * Computer Engineering

Fig. 7: Student Erasmus Application Process #1

In the Erasmus Application form, student name and surname, STARS ID, department are retrieved from the database, requiring no user input. This helps to reduce the complexity of the application process. However, the students can still see these fields to check for wrong data.



Erasmus Application

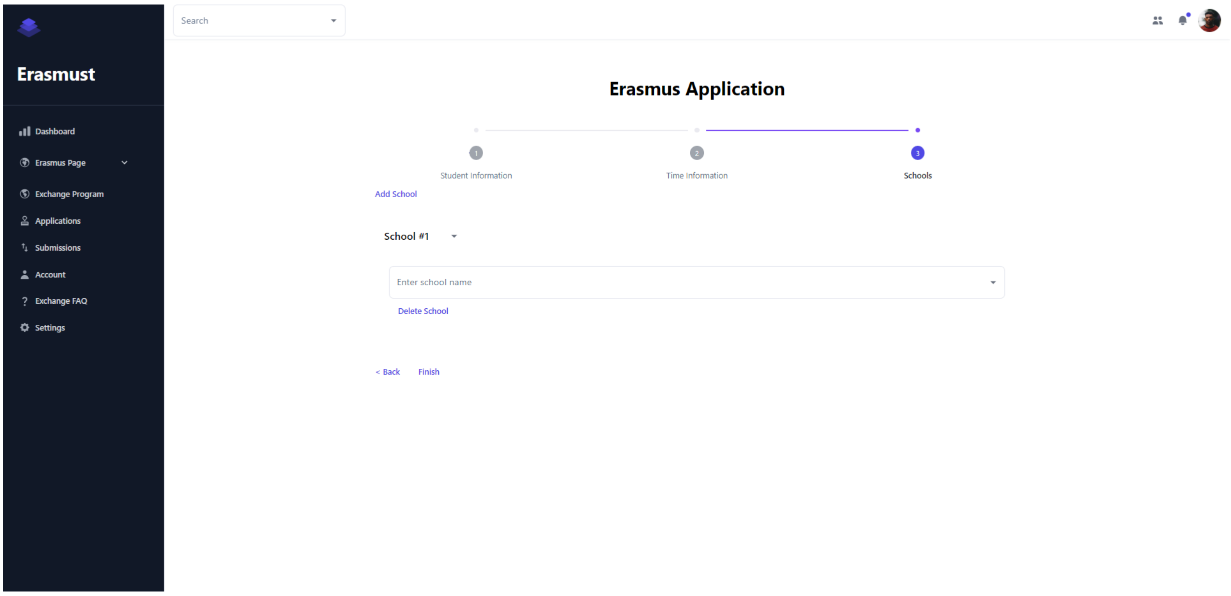
Time Information
Program time information

Academic Year *

Enter semester

Fig. 8: Student Erasmus Application Process #2

In the next stage of the Erasmus Application Form, time requirements are asked from the user. Time requirements are academic year and semester. Enter semester is a dropdown menu and consists of Fall and Spring. These are editable fields, and form validations are done for both of the fields.



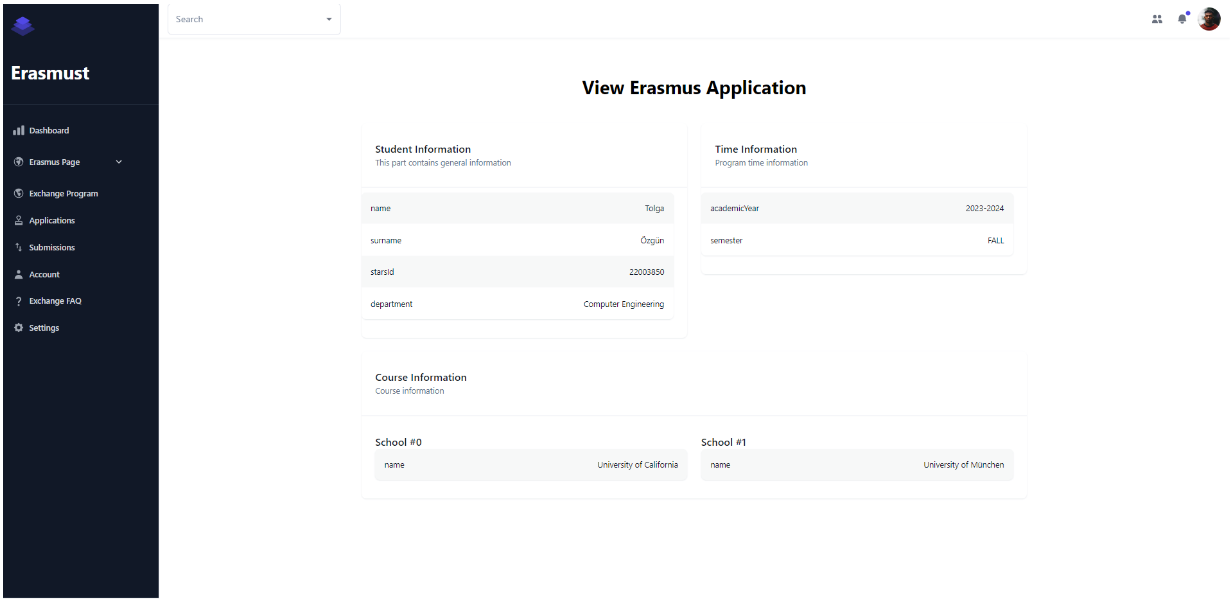
Erasmus Application

School #1

Enter school name

Fig. 9: Student Erasmus Application Process #3

In the next stage of the Erasmus Application Form, student enters their choice of school for application. Schools are retrieved from database and listed in a dropdown menu with auto complete.



View Erasmus Application

Student Information
This part contains general information

name	Tolga
surname	Özgün
starsId	22003850
department	Computer Engineering

Time Information
Program time information

academicYear	2023-2024
semester	FALL

Course Information
Course information

School #0

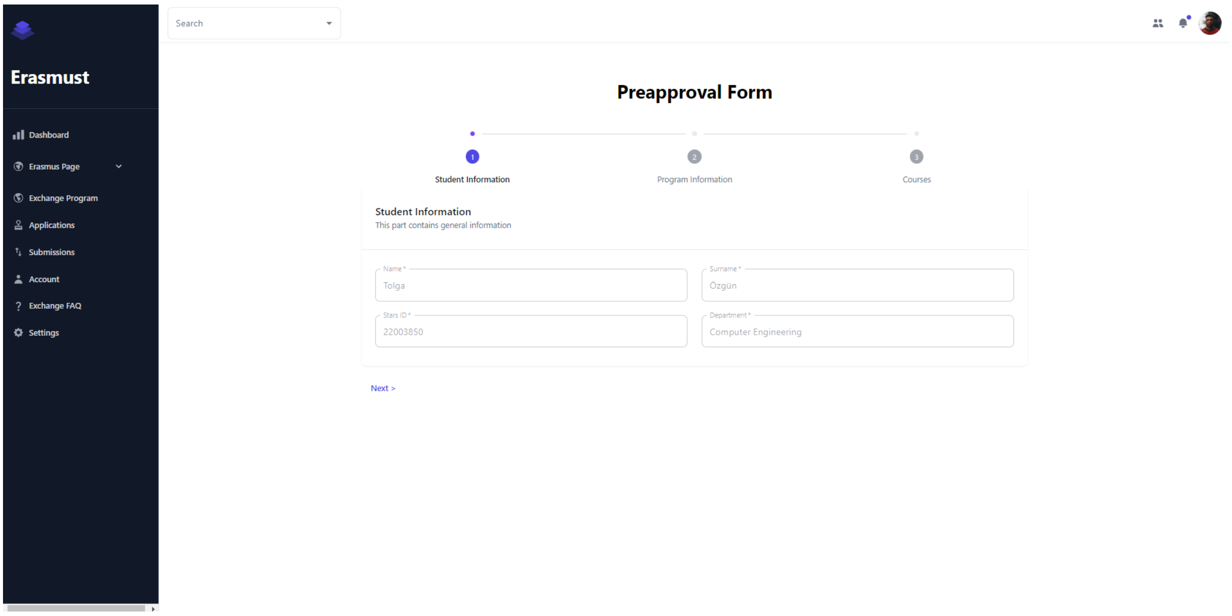
name	University of California

School #1

name	University of München

Fig. 10: Student View Erasmus Application

Students can view their approved Erasmus Application info. These are retrieved from database and are not editable.



Preapproval Form

Student Information

This part contains general information

Name *: Tolga

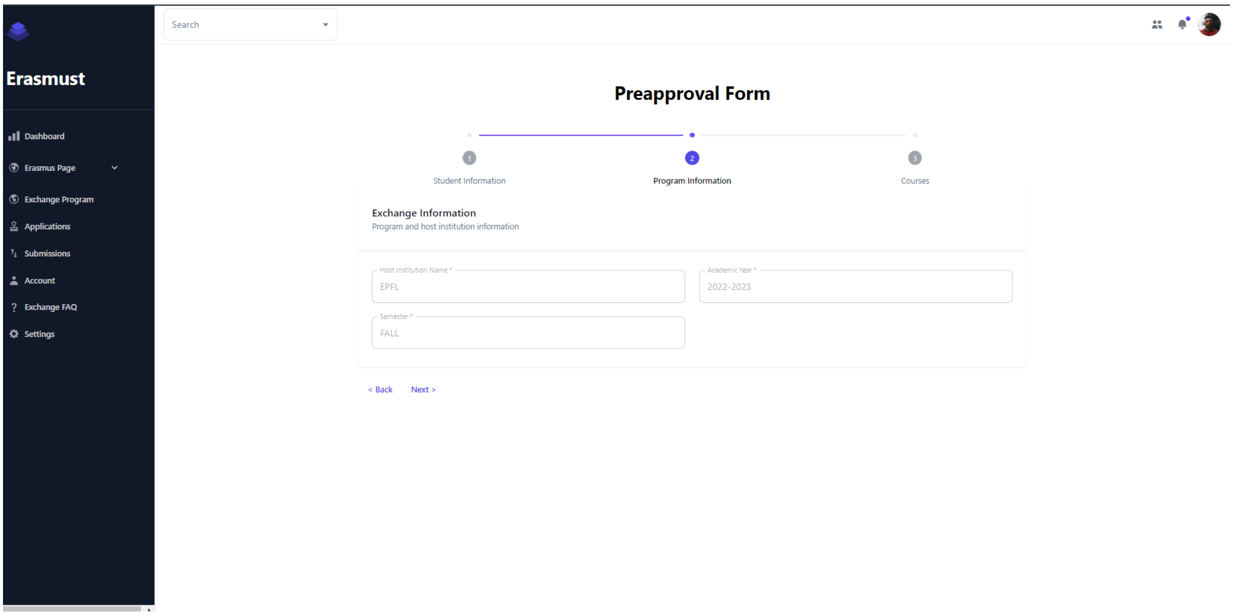
Surname *: Özgün

Stars ID *: 22003850

Department *: Computer Engineering

Fig. 11: Student Preapproval Submission #1

In the Preapproval Form, student name and surname, STARS ID, department are retrieved from the database, requiring no user input. This helps to reduce the complexity of the application process. However, the students can still see these fields to check for wrong data.Student can proceed to next step by clicking Next button.



Preapproval Form

Exchange Information

Program and host institution information

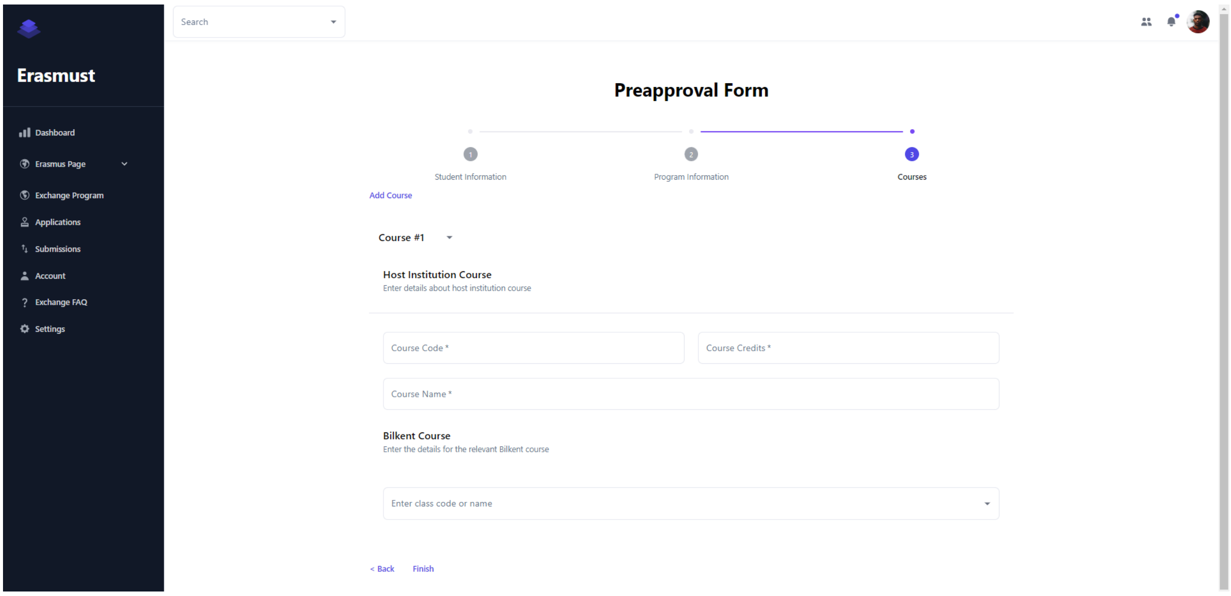
Host Institution Name *: EPFL

Academic Year *: 2022-2023

Semester *: FALL

Fig. 12: Student Preapproval Submission #2

In the next stage of Preapproval Form, Host Institution Name, Academic Year and Semester are retrieved from the backend which was previously requested in the Erasmus Applications Form. These are not editable fields but regardless, student can view these. Student can proceed to the next step by clicking Next button and go back to previous step by clicking Back button.



Preapproval Form

1 Student Information — 2 Program Information — 3 Courses

Add Course

Course #1

Host Institution Course
Enter details about host institution course

Course Code *

Course Credits *

Course Name *

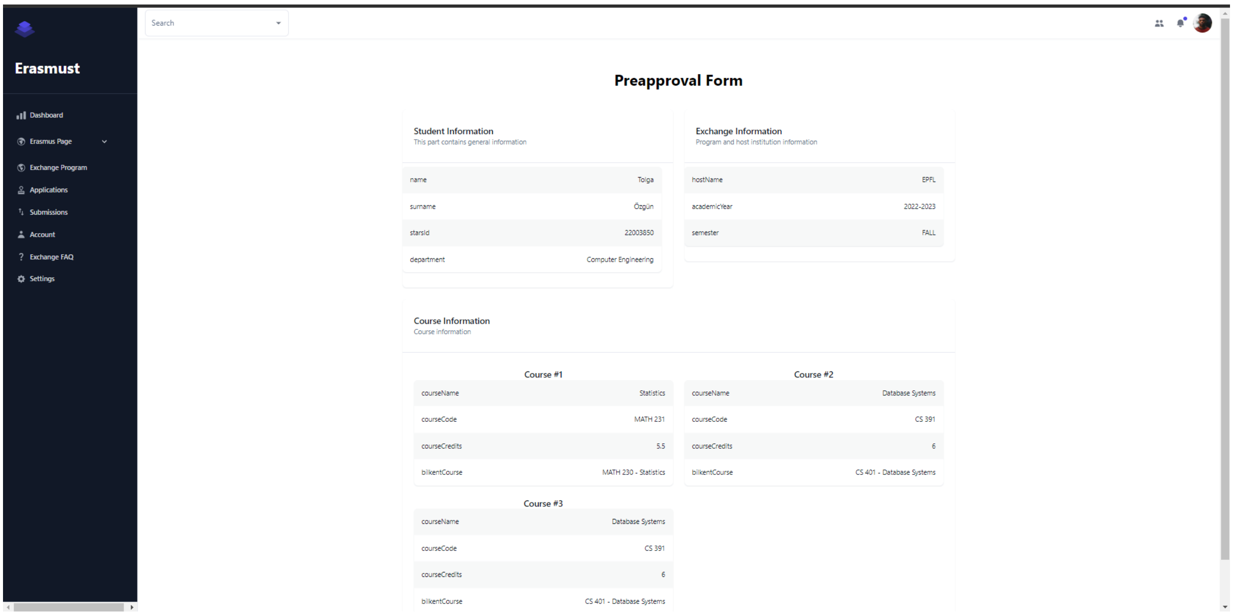
Bilkent Course
Enter the details for the relevant Bilkent course

Enter class code or name

< Back Finish

Fig. 13: Student Preapproval Submission #3

In the next stage of Preapproval Form Student can add a course using Add Course button to add more courses in their Preapproval Form. Each course contains fields such as Host University Course Code, Host University Course Credits, Host University Course Name, and course’s corresponding course in Bilkent University. Students can submit their form by clicking the Finish button.



Preapproval Form

Student Information
This part contains general information

name	Tolga
surname	Özgün
starsId	22003850
department	Computer Engineering

Exchange Information
Program and host institution information

hostName	EPFL
academicYear	2022-2023
semester	FALL

Course Information
Course information

Course #1

courseName	Statistics
courseCode	MATH 231
courseCredits	5.5
bilkentCourse	MATH 230 - Statistics

Course #2

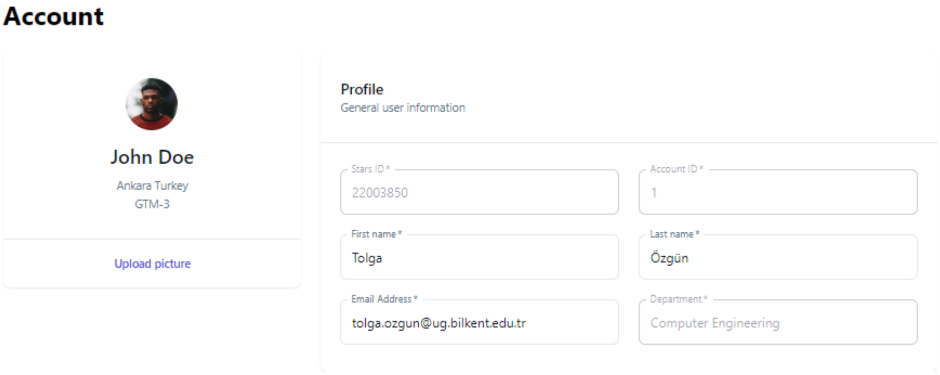
courseName	Database Systems
courseCode	CS 391
courseCredits	6
bilkentCourse	CS 401 - Database Systems

Course #3

courseName	Database Systems
courseCode	CS 391
courseCredits	6
bilkentCourse	CS 401 - Database Systems

Fig. 14: Student View Preapproval Form

Similar to Figure 10, students can view their Preapproval Form info with non editable tables.



Account

John Doe

Ankara Turkey

GTM-3

Upload picture

Profile

General user information

Stars ID * 22003850

Account ID * 1

First name * Tolga

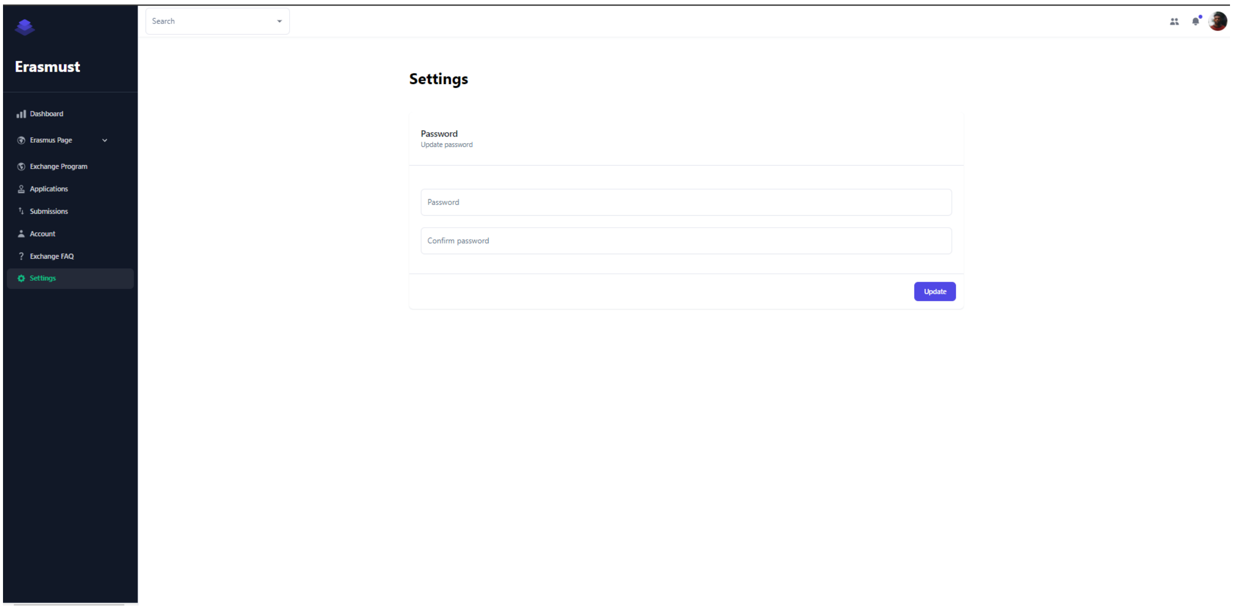
Last name * Özgün

Email Address * tolga.ozgun@ug.bilkent.edu.tr

Department * Computer Engineering

Fig. 15: Student Account

Student can view their Stars Id, Account Id, First Name, Last Name, Email Address and Department. But can only edit their First Name, Last Name and their email address.



Settings

Password

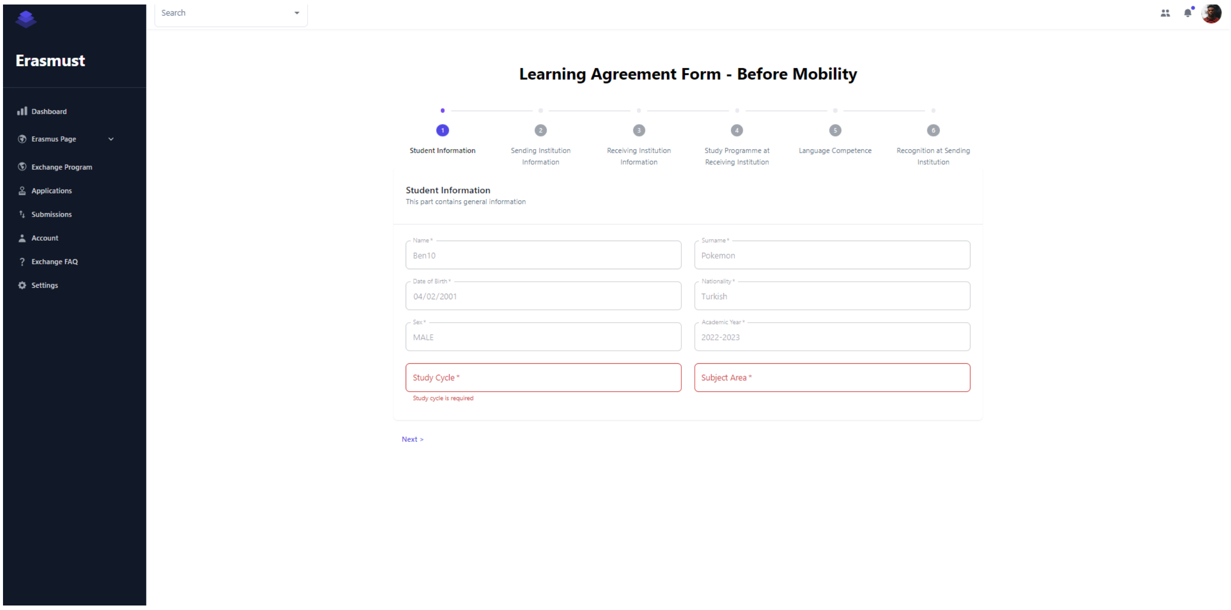
Update password

Password

Confirm password

Fig. 16: Student Settings

Students can change their password. Form validation is done in order to prevent submission if passwords do not match.



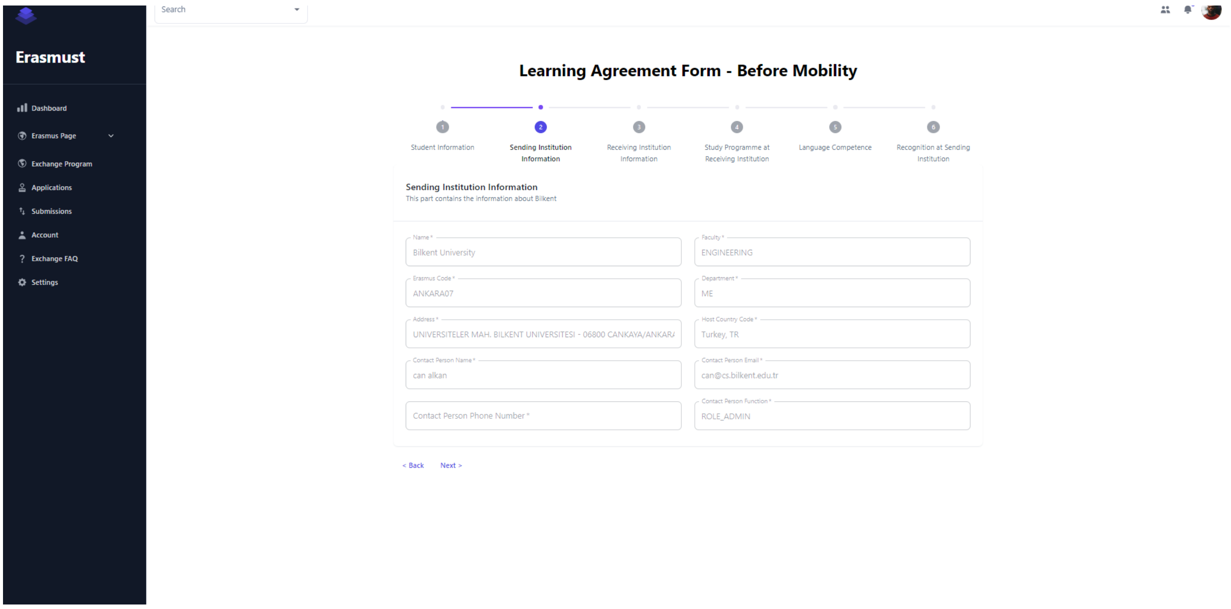
Learning Agreement Form - Before Mobility

Student Information
This part contains general information

Name * Ben10	Surname * Pokemon
Date of Birth * 04/02/2001	Nationality * Turkish
Sex * MALE	Academic Year * 2022-2023
Study Cycle * Study cycle is required	Subject Area *

Fig. 17: Student Learning Agreement Submission - Before Mobility #1

In the Learning Agreement Form, student name and surname, surname, date of birth, nationality, sex, Academic year, are retrieved from the database, requiring no user input. This helps to reduce the complexity of the application process. However, the students can still see these fields to check for wrong data. Study cycle and Subject Area are required fields for student to enter. Student can proceed by clicking next button.



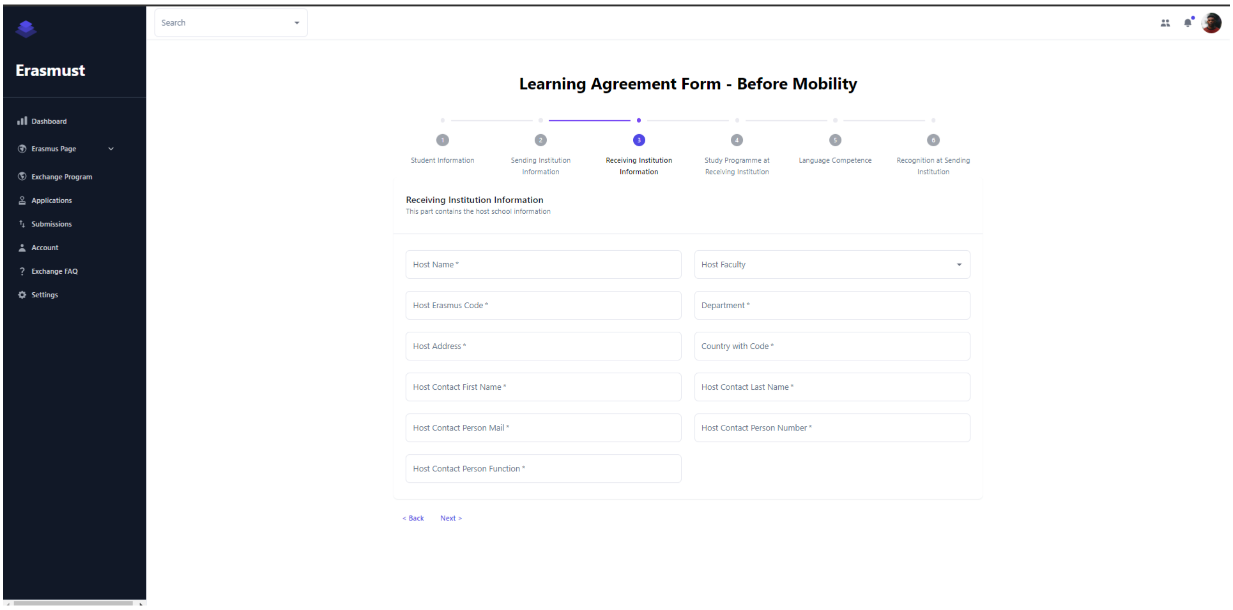
Learning Agreement Form - Before Mobility

Sending Institution Information
This part contains the information about Bilkent

Name * Bilkent University	Faculty * ENGINEERING
Erasmus Code * ANKARA07	Department * ME
Address * UNIVERSITELER MAH. BILKENT UNIVERSITESI - 06800 CANKAYA/ANKAR/	Host Country Code * Turkey, TR
Contact Person Name * can alkan	Contact Person Email * can@cs.bilkent.edu.tr
Contact Person Phone Number *	Contact Person Function * ROLE_ADMIN

Fig. 18: Student Learning Agreement Submission - Before Mobility #2

In the next stage of Learning Agreement Form Sending Institution Informations are filled from the database as it requests knowledge about Bilkent University. Course coordinator details are pulled from previous applications.



Learning Agreement Form - Before Mobility

Student Information | Sending Institution Information | Receiving Institution Information | Study Programme at Receiving Institution | Language Competence | Recognition at Sending Institution

Receiving Institution Information

This part contains the host school information

Host Name *

Host Faculty

Host Erasmus Code *

Department *

Host Address *

Country with Code *

Host Contact First Name *

Host Contact Last Name *

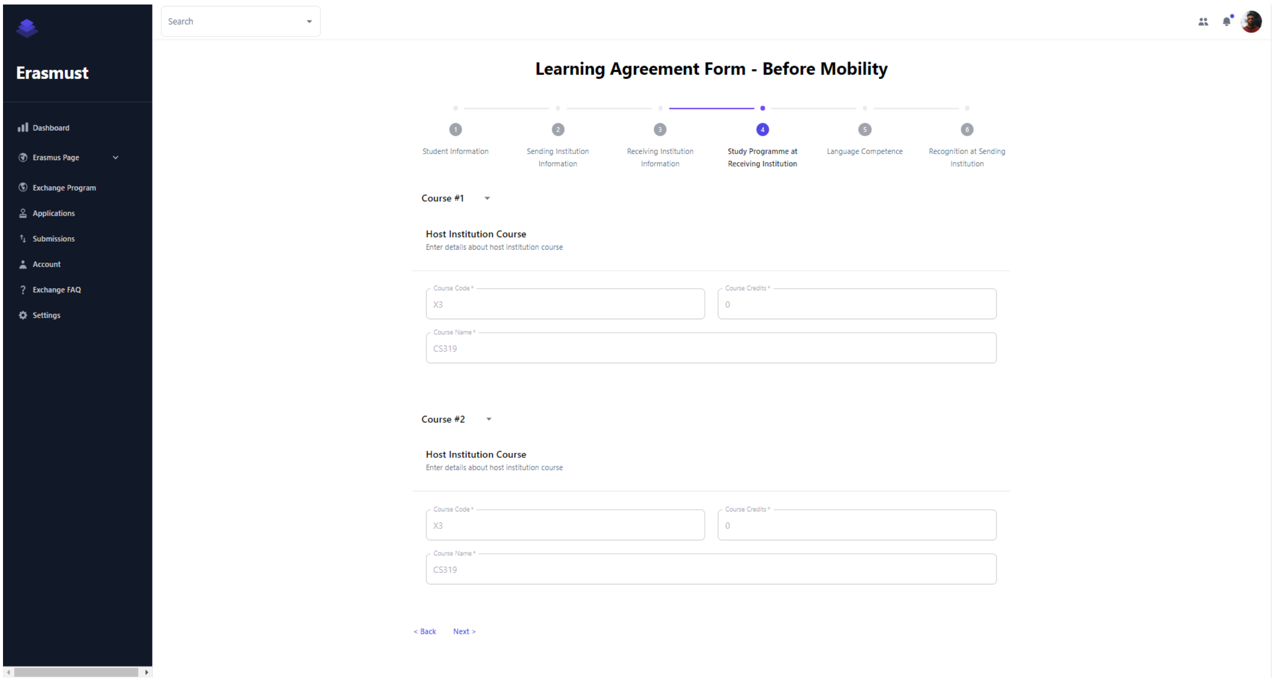
Host Contact Person Mail *

Host Contact Person Number *

Host Contact Person Function *

Fig. 19: Student Learning Agreement Submission - Before Mobility #3

In the next stage of Learning Agreement Form, receiving instution informations are requested. These are; Host Name, Host Faculty, Host Erasmus Code, Department, Host Adress, Country with Code, Host Contact First Name, Host Contact Last Name, Host Contact Person Mail, Host Contact Person Number, Host Contact Person Function. Student can submit this and proceed to next step by next button.



Learning Agreement Form - Before Mobility

Student Information | Sending Institution Information | Receiving Institution Information | Study Programme at Receiving Institution | Language Competence | Recognition at Sending Institution

Course #1

Host Institution Course

Enter details about host institution course

Course Code *: X3

Course Credits *: 0

Course Name *: CS319

Course #2

Host Institution Course

Enter details about host institution course

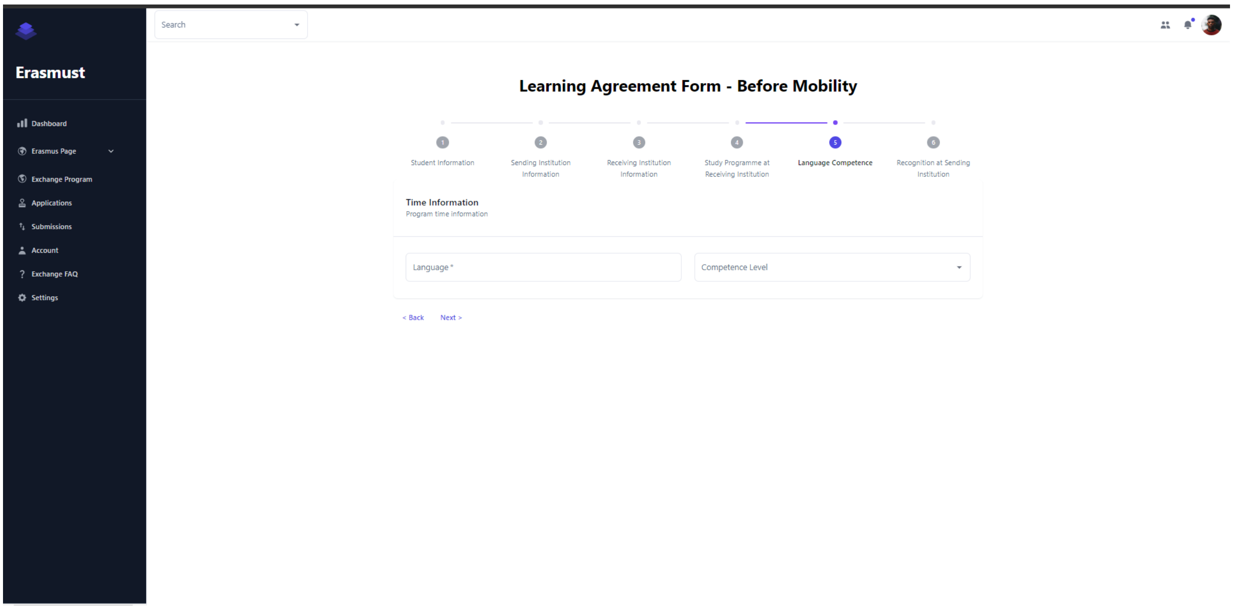
Course Code *: X3

Course Credits *: 0

Course Name *: CS319

Fig. 20: Student Learning Agreement Submission - Before Mobility #4

In the fourth stage of Learning Agreement, courses are automatically retrieved from the approved Preapproval forms and Course Review Forms. User cannot edit or any courses here, but can view what courses are included in their application.



Learning Agreement Form - Before Mobility

1 Student Information
2 Sending Institution Information
3 Receiving Institution Information
4 Study Programme at Receiving Institution
5 Language Competence
6 Recognition at Sending Institution

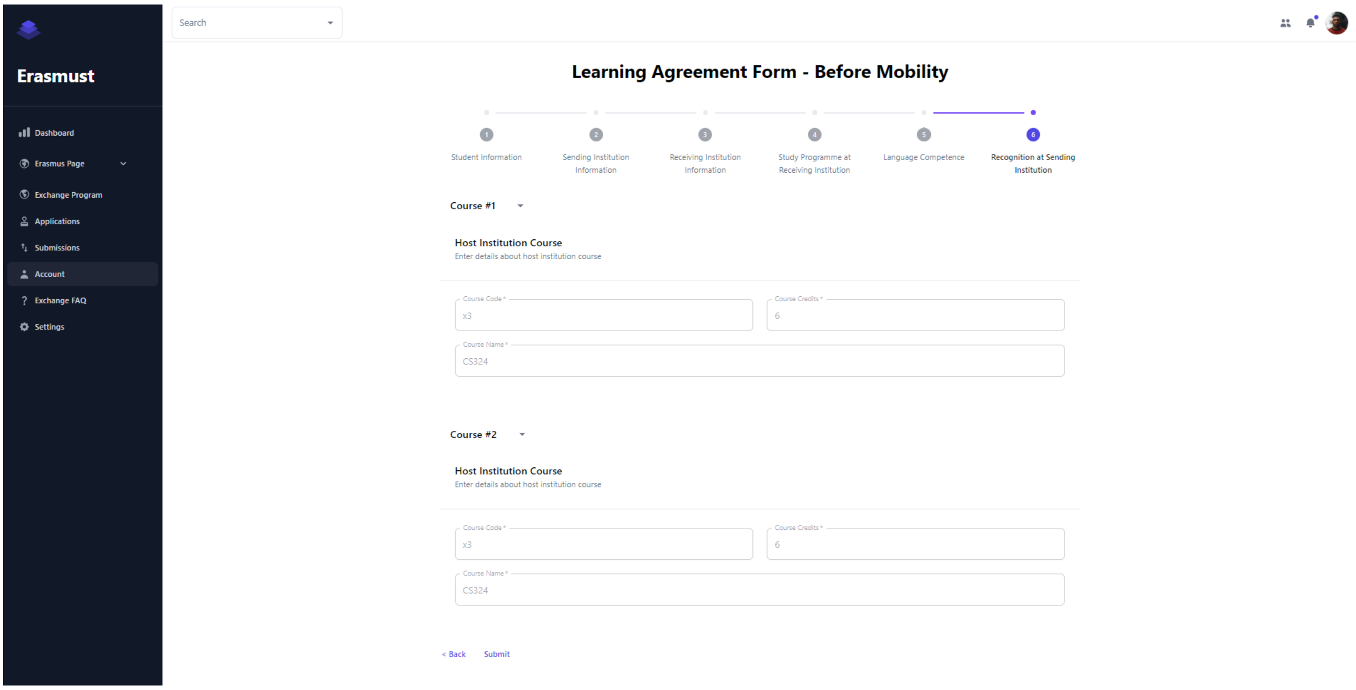
Time Information
Program time information

Language *
Competence Level

< Back Next >

Fig. 21: Student Learning Agreement Submission - Before Mobility #5

In the next stage of Learning Agreement Language Competence fields are requested from user. These are Language and Competence level, this is a drop down menu like A1,A2,B1, etc.



Learning Agreement Form - Before Mobility

1 Student Information
2 Sending Institution Information
3 Receiving Institution Information
4 Study Programme at Receiving Institution
5 Language Competence
6 Recognition at Sending Institution

Course #1

Host Institution Course
Enter details about host institution course

Course Code *: x3
Course Credits *: 6
Course Name *: CS324

Course #2

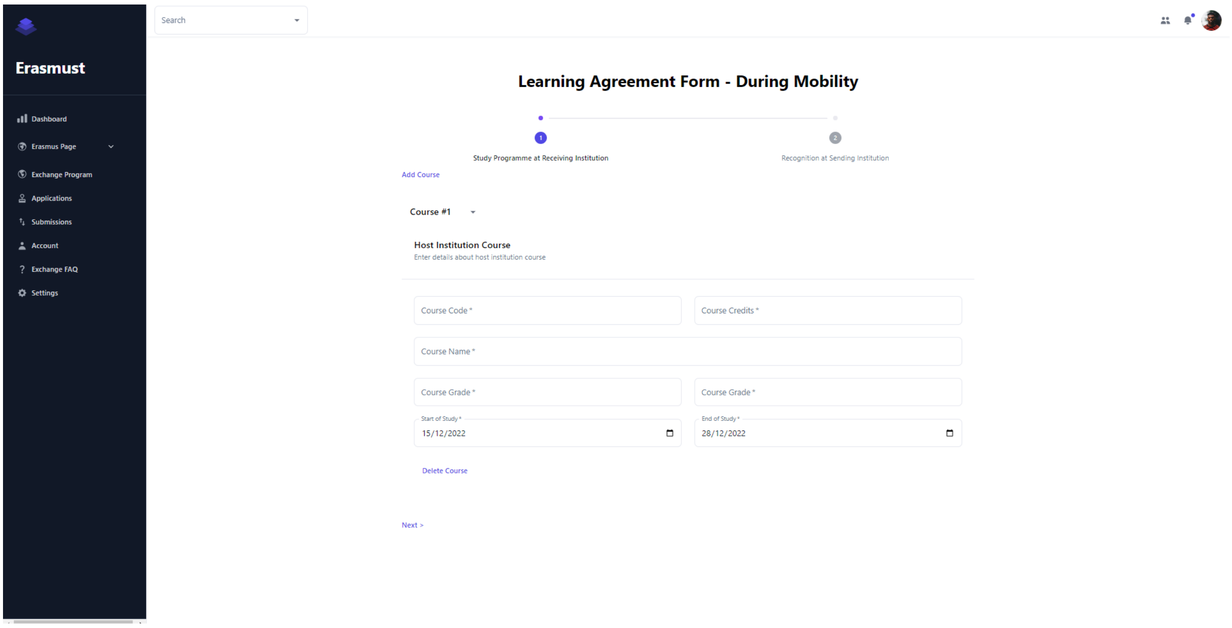
Host Institution Course
Enter details about host institution course

Course Code *: x3
Course Credits *: 6
Course Name *: CS324

< Back Submit

Fig. 22: Student Learning Agreement Submission - Before Mobility #6

In the sixth stage of Learning Agreement, recognition of courses at the host university are automatically retrieved from the approved Preapproval forms and Course Review Forms. User cannot edit or any courses here, but can view what courses are included in their application.



Learning Agreement Form - During Mobility

1 Study Programme at Receiving Institution

2 Recognition at Sending Institution

Add Course

Course #1

Host Institution Course

Enter details about host institution course

Course Code *

Course Credits *

Course Name *

Course Grade *

Course Grade *

Start of Study *: 15/12/2022

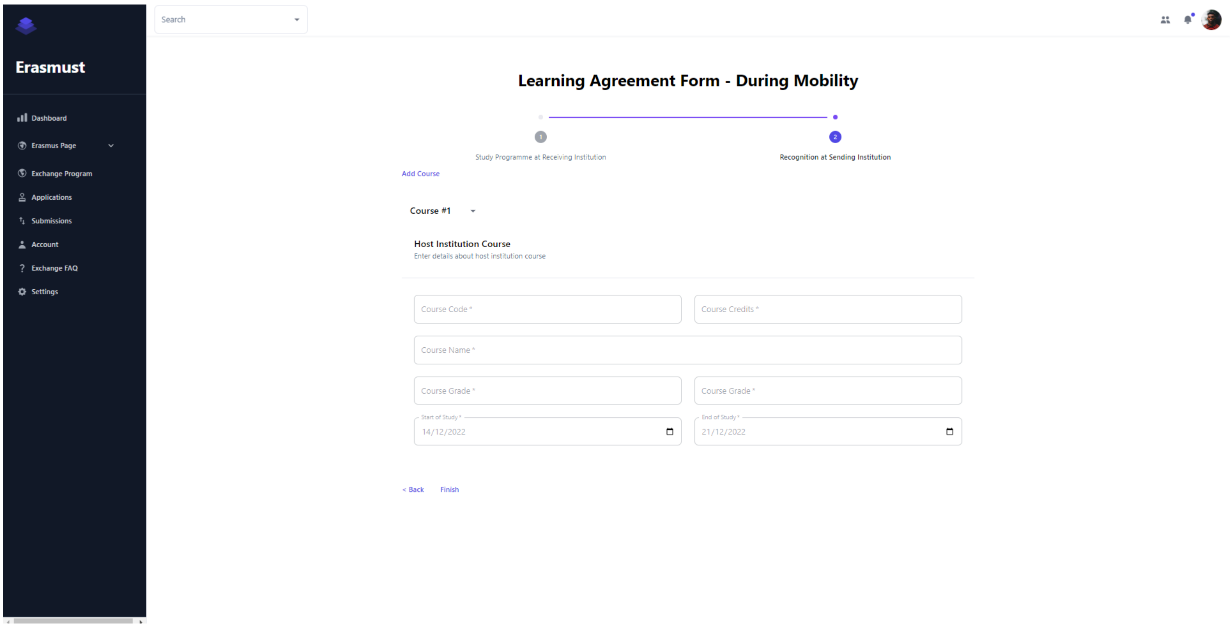
End of Study *: 28/12/2022

Delete Course

Next >

Fig. 23: Student Learning Agreement Submission - During Mobility #1

Students can add exceptions to their Learning Agreements during their mobility period, changing their course with a reason and adding a new course in replacement. In first step, the student can enter a course that they will take at the host university instead of their proposed course.



Learning Agreement Form - During Mobility

1 Study Programme at Receiving Institution

2 Recognition at Sending Institution

Add Course

Course #1

Host Institution Course

Enter details about host institution course

Course Code *

Course Credits *

Course Name *

Course Grade *

Course Grade *

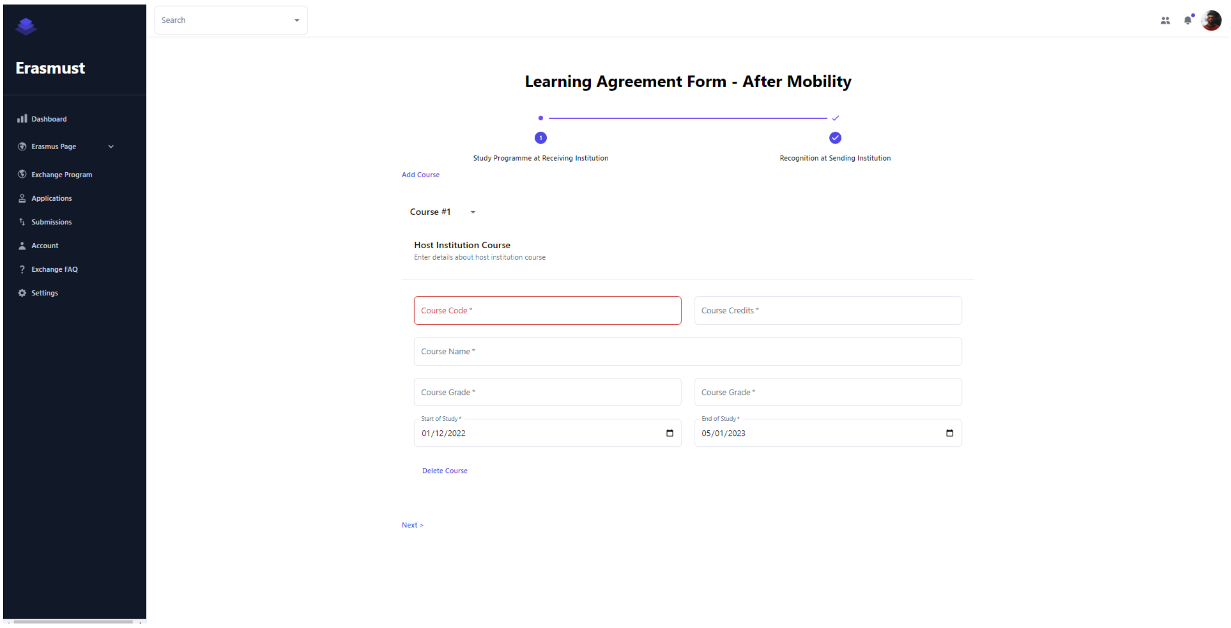
Start of Study *: 14/12/2022

End of Study *: 21/12/2022

< Back Finish

Fig. 24: Student Learning Agreement Submission - During Mobility #2

In second step, the student can change the course's equivalent in Bilkent. Changes in both steps will automatically be applied to their Learning Agreement.



Learning Agreement Form - After Mobility

1 Study Programme at Receiving Institution

✓ Recognition at Sending Institution

Course #1

Host Institution Course

Enter details about host institution course

Course Code *

Course Credits *

Course Name *

Course Grade *

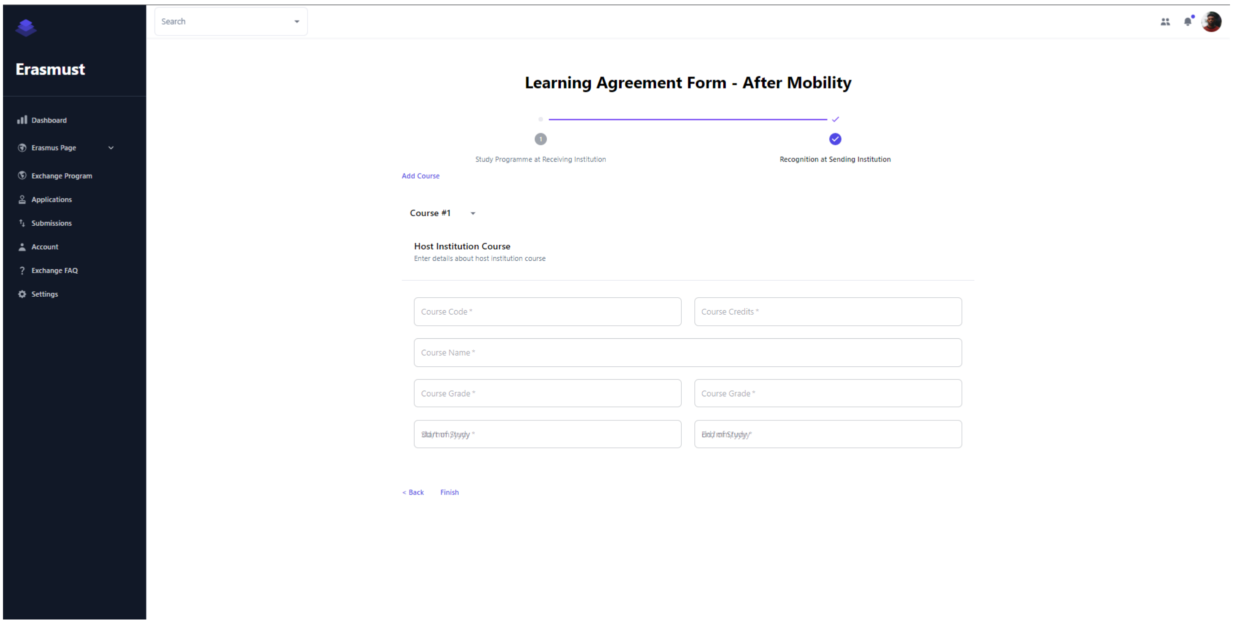
Course Grade *

Start of Study * 01/12/2022

End of Study * 05/01/2023

Fig. 25: Student Learning Agreement Submission - After Mobility #1

In the Learning Agreement Form After Mobility Erasmus Coordinatoris requested to enter course codes that he has taken in the host university and their corresponding credits grade.



Learning Agreement Form - After Mobility

1 Study Programme at Receiving Institution

✓ Recognition at Sending Institution

Course #1

Host Institution Course

Enter details about host institution course

Course Code *

Course Credits *

Course Name *

Course Grade *

Course Grade *

Start of Study *

End of Study *

Fig. 26: Student Learning Agreement Submission - After Mobility #2

In the Learning Agreement Form After Mobility Erasmus Coordinator is requested to enter course codes that he has taken in the host university and their corresponding credits grade.

4. Build Instructions

One can either use the pre-existing application, or build it from source files to host it. As the application is currently up and running in a stable environment, anyone can use the application to their needs.

However, if one wants to re-host the application, they can do this by building on their machines, but by giving credits to BubbleZort team.

4.1. Using the application

Erasmust application is up and running at https://myerasmust.com. As an alternative link one can use (http://92.205.25.135:2) in the web browser. This application is available to use and is stable.

4.2. Building the application

1. Pull the project code from https://github.com/tolgaozgun/erasmust
2. Install Docker to your machine. Docker currently supports Linux, Mac and Windows. Following is the official documentation for installation: https://docs.docker.com/engine/install/
3. Build Docker images using the docker-compose.yml file. This file is located on the main directory. Point Docker to main directory and build using compose file. Following is the official documentation for building using compose file: https://docs.docker.com/engine/reference/commandline/compose_build/
4. Use the compose file to run the servers. Docker will automatically install the dependencies and run the server for you. Additionally, one can configure the ports in docker-compose.yml to specify endpoints in the web browser. After a successful run, docker logs will give you the URL to browse to open the application. Follow the official documentation to run: https://docs.docker.com/engine/reference/commandline/compose_up/

5. Work Allocation

- **Eylül Badem**

Created activity diagrams for course transfer and application evaluation processes, and wrote general explanations for Requirement Analysis Report. Created the Entity Diagram and wrote some general explanations for the Design Report. Took part in the creation of the back-end structure, mainly worked on Learning Agreement Form in the implementation. Did the necessary footage and editing for the demo video.

- **Tolga Özgün**

Created CI/CD pipeline using an existing Virtual Dedicated Linux Server hosted on GoDaddy.com, Dockerized the Spring, React and Postgre applications. Created frontend designs for all forms. Wrote a small amount of Spring services to pass data to frontend. Took an essential part in creating many frontend components such as navbar, search, router. Created use case diagram with textual descriptions and management layer. Wrote boundary conditions, build instructions in reports.

- **Yahya Eren Demirel**

Took part in the design of uml class and database diagrams during the reporting and planning process. In the implementation process, took part in the creation of the back-end package architecture, the integration of spring security, and the design of the pre-approval form and course review forms.

- **Nisa Yılmaz**

  Created application and preapproval activity diagrams and wrote general explanations in requirements report. Wrote management layer explanations and managed general report formatting in design report. Implemented erasmus application and course transfer in backend implementation, helped with overall structure of backend and wrote getter methods to establish communication with frontend.

- **Emirhan Büyükkonuklu**

  Created state diagrams for Incoming Student, Outgoing Student, Course Coordinator; and Erasmus/Exchange Coordinator Activity Diagram, in Requirements Report. Created Interface Layer Diagram with Barış and their corresponding comments in Design Report. In frontend, participated mostly in writing static view pages. Took part in navbar and router. Implemented Todo list for staff.

- **Barış Yıldırım**

  Created sequence diagram, and wrote explanation for it in requirements report. Created interface layer diagram in design report. Created most of the frontend view, edit and list pages for student and admin pages. Took part in frontend components such as sidebar, file uploads and list components.

6. What We Did

We could do with almost no bugs:

- Creating, listing and editing Erasmus application
- Creating, listing and editing Preapproval for Erasmus application
- Creating, listing and editing Learning Agreement for Erasmus application
- Creating, listing and editing Course Transfer forms
- Creating, listing and editing Course Review forms
- Uploading PDF files with Course Review forms.
- Prefilling Learning Agreement form data from Erasmus Application and Preapproval Application.
- Prefilling Preapproval Application data from Preapproval Application
- Giving the option for course coordinators to include personalized message that will be shown to user during the Course Review Form creation process.
- Create Erasmus points for each user based on the calculation formula on Erasmus page.
- Place students in universities based on their Erasmus points
- Add host schools specific to departments.
- UI Navigation
- Profile Page
- Generating PDF files from forms.

We could not manage to finish:

- Exchange Applications
- Chat
- Notifications
- Incoming Student
- Two Factor Authentication