



Bilkent University

Department of Computer Engineering

# CS 319 TERM PROJECT

## Section 2

*ErasmusAga*

Group 2B

**Project name: Easy Erasmus**

## Project Analysis Report - Iteration 2

Doruk Altan 21401362 [doruk.altan@ug.bilkent.edu.tr](mailto:doruk.altan@ug.bilkent.edu.tr)

Asım Bilal Ak 21802887 [bilal.ak@ug.bilkent.edu.tr](mailto:bilal.ak@ug.bilkent.edu.tr)

Gökberk Altıparmak 21901798 [g.altiparmak@ug.bilkent.edu.tr](mailto:g.altiparmak@ug.bilkent.edu.tr)

Furkan Yıldırım 21902514 [f.yildirim@ug.bilkent.edu.tr](mailto:f.yildirim@ug.bilkent.edu.tr)

Alp Tuğrul Ağçalı 21801799 [tugrul.agcali@ug.bilkent.edu.tr](mailto:tugrul.agcali@ug.bilkent.edu.tr)

Instructor: Eray Tüzün

Teaching Assistant: İdil Hanhan

# Contents

<b>1 Introduction</b>	<b>4</b>
<b>2 Overview</b>	<b>5</b>
<b>2.1 Student</b>	<b>5</b>
2.1.1 Features Of Students	5
2.1.1.1 Profile of Students	5
2.1.1.2 Grades Of Students	5
2.1.1.3 Applications of Students	6
2.1.1.4 Course Selection Of Student's	6
2.1.2 Authority Of Students	6
2.1.2.1 Students will be able to check information about their Application	6
2.1.2.2 Students will be able to cancel their Erasmus application	7
2.1.2.3 Students will be able to upload file to the application	7
<b>2.2 Erasmus Administrators</b>	<b>7</b>
2.2.1 Erasmus Administrators will be able to see the student's personal information	7
2.2.2 Erasmus Administrators will be able to see the applied universities for Erasmus by student	7
2.2.3 Erasmus Administrators will be able to see the the wanted courses to be taken by students at pre-approval stage	8
2.2.4 Erasmus Administrators will have a to-do list	8
2.2.5 Erasmus Administrators will confirm the erasmus application at the end of process	8
<b>2.3 Course Coordinators</b>	<b>8</b>
2.3.1 Course Coordinators will be able to check whether demanded course is matched with any course in department	8
<b>2.4 Admins</b>	<b>9</b>
2.4.1 Admins will be able to register new users and deregister existing users	9
2.4.2 Admins will be able to assign roles to the users	9
2.4.3 Admins will be able to add new university or delete the existing university	9
2.4.4 Admins will be able to create new entities or delete the existing entities	10
<b>2.5 Universities</b>	<b>10</b>
2.5.1 Universities will have a course offering list that students can view	10
2.5.2 Universities will have an information page that shows additional or different requirements than Bilkent	10
2.5.3 University requirements can be changed	11

<b>3 Proposed System</b>	11
<b>3.1 Functional Requirements</b>	11
3.1.1 Log in	11
3.1.2 Importing data from local excel file	11
3.1.3 Fetching the information from excel file which consist of students information to database	12
3.1.4 Automatically determining universities for students	12
3.1.5 Course approval	12
3.1.6 Uploading pre approval form	13
3.1.7 Checking Language & Semester Constraints	13
<b>3.2 Additional Requirements</b>	13
3.2.1 To-Do List	13
3.2.2 Approval Status For Submissions	14
<b>3.3 Non-functional Requirements</b>	14
3.3.1 Usability	14
3.3.2 Reliability	14
3.3.3 Supportability	15
<b>4 System Models</b>	15
<b>4.1 Scenarios</b>	15
4.1.1 Use-Case Model	22
4.1.2 Object and Class Model	23
<b>4.2 Dynamic Models</b>	23
4.2.1 Sequence Diagrams	23
4.2.2 State Diagrams	27
4.2.3 Activity Diagrams	29
<b>5 User Interface</b>	31
<b>6 References</b>	43

# 1 Introduction

This project aims to create a web application for the erasmus system of Bilkent University. This system is designed to give students the chance to study and live abroad at the universities that Bilkent has contracted with. The project mainly aims to help the erasmus application process for both the students and the instructors by decreasing the workload and speed up the work for them to deal with the process of applications of students. Through this report, models, classes, objects and users of the project will be explained in detail. In order to provide a more useful website for users, we as a group planned to give both students and instructors more opportunity to communicate with them easily.

PostgreSQL will be used for DataBase during the development phase of this project. The software language we plan to use is java 8 and the framework we will use is Cuba Platform. The reason why we do not use JMX, the new version of Cuba Platform, is that it is very new and Cuba Platform's forums are more active.

In order to achieve these goals, the features that the program should have were considered in detail by our group. Before explaining the features in more detail, the features that the program will have are as follows:

- Login System
- Data Import From Local Excel File
- Registration Of Courses Approved By The Host University And Adding New Ones
- Pre Approval (Student will choose the courses)
- Uploading Necessary Files
- Communication Between Users via Comments
- E-mail Notifications
- To-Do list for Erasmus administrator

- Canceling Erasmus application
- New Student Acceptance From Backup List

## **2 Overview**

In this web application we have several different user types such as, Erasmus administrators, students, admins and course coordinators. All different types of users have several different functionalities they can use in the application. These different types of users can communicate via application.

### **2.1 Student**

In the erasmus application system students have many functionalities such as uploading required files, seeing status of their erasmus application, communicating with Erasmus administrators and course coordinators via comments. In addition to this, students will be able to see the required documents on their account.

#### **2.1.1 Features Of Students**

In the application, the users in the Student group need to have some features. These features are the features that make them unique, the features that determine their ranking in the application approval, and the features that students should have during the acceptance phase of the courses after the application.

##### **2.1.1.1 Profile of Students**

One of the features that every student has is their profile. Their profiles contain unique information provided by the school, such as students' names and surnames, ID numbers and departments.

##### **2.1.1.2 Grades Of Students**

This section contains information previously provided by Bilkent University to put students in order of success. Both students and Erasmus administrators can

control this section, but cannot edit it. If there is an error in the imported information, it is forwarded to the Erasmus administrator by the student and edited by the admin of the application.

### **2.1.1.3 Applications of Students**

There will be a few features in the students' application feature. One of these features will be the Preferred Duration. Preferred Duration is the feature that the student specifies in which education period he or she prefers to use exchange programs. Another feature of the application is that it keeps a list of applied universities. A student can apply to five universities at the same time. Thanks to the state property of applications, the application's situation can be seen. (accepted, being evaluated, in reserve, denied).

### **2.1.1.4 Course Selection Of Student's**

If the student's Erasmus application is accepted, the student will be able to edit this feature. Here he will be able to see and select previously approved courses provided by the University. In addition, the Student will be able to apply for some courses to be recognized as equivalent.

### **2.1.2 Authority Of Students**

There are some actions that users with the student role can do in the application.

#### **2.1.2.1 Students will be able to check information about their Application**

In this application, students will be able to see a screen with information about their application. On this screen, they will have the opportunity to see not only the application but also their profile information. If there is an error about the name, ID or notes, they will be able to report it to the Erasmus administrator by commenting. They will be able to upload the necessary files here. There will be a checklist here to prevent it from uploading files incompletely.

### **2.1.2.2 Students will be able to cancel their Erasmus application**

For any reason, if the student changed his mind to not go to the university where his/her application is already confirmed, he will have a chance to cancel his application.

### **2.1.2.3 Students will be able to upload file to the application**

As the necessary requirements of the student's application, students will need to upload a file to the application such as, pre-approval form, syllabus of the course etc. Therefore, they will be able to upload a file to the application.

## **2.2 Erasmus Administrators**

In the Erasmus application, Erasmus administrators have several features to use. They control the applications' status and also according to the program output they will see which student wins which university. In addition to this they can claim some files from students for erasmus requirements. According to their file claim Erasmus administrators can see whether students fulfill the requirements. Also, they can communicate via the comments to warn students to fulfill the requirements.

### **2.2.1 Erasmus Administrators will be able to see the student's personal information**

Erasmus administrators can see the student's information such as their name, surname, student ID, Bilkent mail, department, date of birth, applied universities, cgpa, grades of ENG101 and ENG102 courses.

### **2.2.2 Erasmus Administrators will be able to see the applied universities for Erasmus by student**

We thought that it would be important for Bilkent University to see the demanded universities list by students to collect data. Therefore, they may make an agreement with the most demanded universities to increase the appointment number for future applications.

### **2.2.3 Erasmus Administrators will be able to see the the wanted courses to be taken by students at pre-approval stage**

It would be a good data collection for both Bilkent University and the applied university for future applications.

### **2.2.4 Erasmus Administrators will have a to-do list**

In order to quicken the process, Erasmus administrators will be informed by a to-do list which will help them.

### **2.2.5 Erasmus Administrators will confirm the erasmus application at the end of process**

When all of the requirements are fulfilled, Erasmus administrators will be able to confirm the application if everything is fine.

## **2.3 Course Coordinators**

In Erasmus application course coordinators will have few features to use because course coordinators will check the course of foreign universities whether it is appropriate or not for Bilkent University.

### **2.3.1 Course Coordinators will be able to check whether demanded course is matched with any course in department**

If the desired course of the university is not in the list of courses that are already confirmed by Bilkent University, students will upload the syllabus of that course to the system. The uploaded file will be directed to the matching course coordinator to be evaluated. If the desired course of the other university has any matching for the courses of Bilkent University, the course coordinator will confirm the application and the course will be added to the confirmed courses file. Students will be informed by the result of this process.



## **2.4 Admins**

Users in the admin role in this application have the authority to control the features of other users in the program. For example, they can add an instructor or a student, create an application manually, or change the roles of other users.

### **2.4.1 Admins will be able to register new users and deregister existing users**

Because there is no sign up functionality in our application, admins will control who will have access to the system. Therefore, they can register new users to the system and deregister the existing user when they are done with the Erasmus application.

### **2.4.2 Admins will be able to assign roles to the users**

In our Erasmus application, as it mentioned before there are four types of users which are students, Erasmus administrators, course coordinators and admins. Existing admins will be able to assign roles to the user. In other words, admins will decide who will have access to the application as student, Erasmus administrator, course coordinator and admin.

### **2.4.3 Admins will be able to add new university or delete the existing university**

If Bilkent University makes new agreements with new universities under the Erasmus program, admins of the application will add them to the system. Otherwise, they can delete the university from the application.

#### **2.4.4 Admins will be able to create new entities or delete the existing entities**

Because admins are responsible for the entire application, they will be able to add new entities such as, Student, Erasmus Administrators, Course Coordinators and Universities to the application and wipe existing entities from the application.

### **2.5 Universities**

The university entity has certain features and functionalities that inform students about its rules, courses, application process requirements and teaching language. The university policies, the courses and acceptance requirements can be changed by the admin and Erasmus administrator. Students will be able to view courses that are offered by a university, language requirements if any, grading differences, and more.

#### **2.5.1 Universities will have a course offering list that students can view**

Students will be able to see which courses are available at which universities through this property. Also they will be able to view the prerequisites and its corresponding credit award.

#### **2.5.2 Universities will have an information page that shows additional or different requirements than Bilkent**

Some universities or countries may have additional requirements and regulations for acceptance into the Erasmus program such as language requirements, department quotas, semester requirements or different grading scales. The information above will be available for viewing to students via this page.

### **2.5.3 University requirements can be changed**

Policies and requirements of universities might change overtime. Following any such changes, the university will be able to add, delete and alter any existing requirement. These could be courses, prerequisites, GPA thresholds or language requirements.

## **3 Proposed System**

### **3.1 Functional Requirements**

#### **3.1.1 Log in**

Login will be very easy in our web application. Because students will automatically sign up according to the excel file which is provided by the Erasmus administrator. According to this excel file they have student ID and also the program will provide a password which can be changed by students. The Program will give advice to change the password to personalization.

Erasmus administrators and coordinators have password and username which are given by admins.

#### **3.1.2 Importing data from local excel file**

This excel file will be uploaded by administrators through copying the local path of the excel file to the desired place in the application. And in this excel file, there will be students' names, their ID's, their grades and Universities which they choose for Erasmus. Those Universities will be consecutive according to students' will.

### **3.1.3 Fetching the information from excel file which consist of students information to database**

According to the excel file, the application will create new student entities with their names, ID's and with other properties. In addition to student entities, the program creates applications of the students. According to those entities which are recorded to the database by program, students will be signed up to the program. Moreover, the program will show all students to Erasmus administrators with their information as a list. By only double clicking the student's name, Erasmus administrators will see detailed information about those students.

### **3.1.4 Automatically determining universities for students**

The process of placing students in the universities they deserve, which is one of the main purposes of this application, will be carried out simultaneously with the excel import. After the program creates each application entity, it will automatically check the vacancies in that university and place the student if there is an empty place. If a student's application with a higher score is created later than this application, the previously placed application will be discarded. If the application quota is full, the student who lost his position goes to his next choice. If he cannot settle in his next preferences, he enters the reserves.

If a placed student cancels his/her application, the member in the waiting bin with the highest score is placed.

### **3.1.5 Course approval**

For an exchange between universities, the courses that the student plans to take must be approved. Must courses must be approved by the corresponding Bilkent Instructor, elective courses can be approved by the course coordinators. If a course is already pre-approved, that course is automatically approved, but some courses may not be pre-approved. In order to have these courses approved, they will be able to contact the course coordinator and write a message about why the course should be approved.

### **3.1.6 Uploading pre approval form**

After the courses are approved, students have to prepare the typed pre-approval form. Students must upload their pre-approval forms as a pdf file to the system to apply. This file should be uploaded before the mobility starts. According to this form, the course coordinator checks whether the application is suitable and the application process continues accordingly.

Different universities may have different pre approval forms due to different rules. Students will be able to upload these forms to the system according to these changes. While the pre-approval form is uploaded, the country of the university will be selected and the application form will be treated accordingly.

### **3.1.7 Checking Language & Semester Constraints**

The information about the semester and language requirements for different universities are present in their corresponding information page. However, to eliminate any mishaps, the application will check whether the constraints are met. If a student is on a semester that is ineligible with the target university's policies, an application cannot be made. Likewise, if the language requirements for a university is a different language than English, the application will prompt the student to upload a certification or document to provide proof that they meet the language requirements.

## **3.2 Additional Requirements**

### **3.2.1 To-Do List**

To-do lists appear for any user of the app. It shows the stage of the application for students, and according to this list, students can see how they should proceed, which files they need to upload to the system and when. Course coordinators and Erasmus administrators can see the documents they have received on a single page in their to-do lists. From this page, when they want to process these documents later, they can reach the document with one click and perform the desired action, and if the work is done with this document after the process, they can mark it as done.

### **3.2.2 Approval Status for Submissions**

The main purpose of this page is to show students the application process in one page. The page prompts the stage of each process that students have done, for example, whether the application is being reviewed or not. Wherever possible, the application will also show the underlying processes as in whose approvals are given and whose are left in order to complete the entire approval process. In addition, this page shows an image of the actions that students have done before, and provides information about what the next stage is.

## **3.3 Non-functional Requirements**

### **3.3.1 Usability**

Communications will be operated thanks to the comment system and also mail notifications will be sent to the user so, user can know the change even if they are not in the web platform. Also, Erasmus administrators have to make a list to keep tracing students' duties. In addition to this, Students have a checklist which they must do. Therefore, without physical communication between student and Erasmus administrator both of them will know what to do for requirements of that Erasmus. Moreover, in the program many pieces of information are listed in the data grids and that information includes other information. For example, students should be chosen to see information about students. Choosing can be done by double click to avoid accidentally choosing. It is important because the speed of the program will increase by user since wrong access to the page will be reduced. Moreover, also when the mouse is on that data grid that data grid will be different color so this feature also will prevent the user from accidentally choosing.

### **3.3.2 Reliability**

All documents and requirements must be done for Erasmus. For this purpose in web applications according to the checklist of students there will be an upload

section for that requirement so students can know exactly what to do. Also, students only can see her/him own situations. Complete accession can be done only by admin so any accidental accessibility can be prevented by this feature. Therefore there will not be any confusion.

In addition to those, the theme of the web will be simple and looks like other usual web applications so user confusion because of the view will be prevented thanks to the simple and basic theme.

### **3.3.3 Supportability**

In our web application all operating systems and web browsers which enable access to the Internet can access our platform.

## **4 System Models**

### **4.1 Scenarios**

#### **Use Case #1**

Use case: View demanded courses

Actors: Erasmus administrator

Entry Conditions:     - Logged in as Erasmus administrator

Exit Conditions:       - Erasmus administrator has access to student's demanded courses

Flow of Events:

- 1) Erasmus administrator selects view student profile for a student from list
- 2) Erasmus administrator accesses student information

#### **Use Case #2**

Use case: View applied universities

Actors: Erasmus administrator

Entry Conditions: - Logged in as Erasmus administrator

Exit Conditions: - Erasmus administrator has access to student's applied universities list

Flow of Events:

- 1) Erasmus administrator selects view student profile for a student from list
- 2) Erasmus administrator accesses applied universities list

### **Use Case #3**

Use case: Views student approval status

Actors: Erasmus administrator

Entry Conditions: - Logged in as Erasmus administrator

Exit Conditions: - Erasmus administrator has access to student's approval status

Flow of Events:

- 1) Erasmus administrator selects view student profile for a student from list
- 2) Erasmus administrator accesses student approval status

### **Use Case #4**

Use case: Confirm student application

Actors: Erasmus administrator

Entry Conditions: - Logged in as Erasmus administrator  
- Student has pre-approved courses

Exit Conditions: - Student application is confirmed

Flow of Events:

- 1) Erasmus administrator selects student from list
- 2) Erasmus administrator checks application status



- 3) Erasmus administrator clicks confirm student application
- 4) Student is notified

### **Use Case #5**

Use case: Cancel application

Actors: Student

Entry Conditions: - Student has a placement

Exit Conditions: - Student no longer has a placement

Flow of Events:

- 1) Student chooses to cancel application
- 2) Student placement is undone
- 3) Student application status is updated
- 4) Student is placed in the waiting bin

### **Use Case #6**

Use case: Upload file

Actors: Student, Erasmus administrator

Entry Conditions: - Actor has a supported file type (i.e. pdf, xls)

Exit Conditions: - File is uploaded to system

Flow of Events:

- 1) Actor clicks upload file
- 2) Actor clicks select file
- 3) Actor enters path or drags file to upload file box
- 4) Actor clicks submit
- 5) Appropriate successful / unsuccessful message is displayed

### **Use Case #7**

Use case: Download file

Actors: Student, Coordinator, Erasmus administrator, Admin

Entry Conditions: - Files list is not empty

Exit Conditions: - Actor has local copy of the file

Flow of Events:

- 1) Actor selects files tab from dashboard
- 2) Actor selects the desired file from the list
- 3) Actor clicks download
- 4) File is downloaded

### **Use Case #8**

Use case: View university course offerings

Actors: Student

Entry Conditions: - none

Exit Conditions: - Student clicks exit

Flow of Events:

- 1) Student selects a university from universities list
- 2) Student clicks view offered course
- 3) Offered courses for that university are displayed

### **Use Case #9**

Use case: Edit To-Do List

Actors: Student, Erasmus administrator

Entry Conditions: - none

Exit Conditions: - Actor clicks save changes

Flow of Events:

- 1) Actor clicks view To-Do list

2) Actor clicks edit To-Do list

3) Actor makes changes

4) Actor clicks save changes

#### **Use Case #10**

Use case: View university requirements

Actors: Student, Erasmus administrator

Entry Conditions: - none

Exit Conditions: - Actor clicks exit

Flow of Events:

- 1) Actor selects a university from universities list
- 2) University information is displayed including requirements
- 3) Actor presses exit

#### **Use Case #11**

Use case: Login

Actors: Student, Coordinator, Erasmus administrator, Admin

Entry Conditions: - Actor has a valid account in database

Exit Conditions: - Actor is logged into system

Flow of Events:

- 1) Actor enters credentials in respective text fields
- 2) System checks if there is a matching account
- 3) Actor is directed to the respective main page according to his role

#### **Use Case #12**

Use case: View approved courses list

Actors: Student, Coordinator, Erasmus administrator, Admin

Entry Conditions: - none

Exit Conditions: - Actor clicks exit

Flow of Events: 1) Actor selects view approved courses list from dashboard  
2) Approves courses list is displayed  
3) Actor clicks exit

### **Use Case #13**

Use case: Make comments

Actors: Student, Coordinator, Erasmus administrator, Admin

Entry Conditions: - none

Exit Conditions: - Actor clicks save comment

Flow of Events: 1) Actor types the comment in the text field  
2) Actor clicks save comment  
3) Erasmus administrator is notified that there is a new comment

### **Use Case #14**

Use case: Change password

Actors: Student, Coordinator, Erasmus administrator, Admin

Entry Conditions: - none

Exit Conditions: - User account in database is updated

Flow of Events: 1) Actor goes to his profile page  
2) Actor clicks edit profile  
3) Actor types the new password in the respective text field  
4) Actor clicks save changes  
5) Password is updated

### **Use Case #15**

Use case: Add course to accepted courses list

Actors: Coordinator

Entry Conditions: - Course must have a syllabus attached to request

Exit Conditions: - Accepted courses list is updated

Flow of Events:

- 1) Coordinator selects the files tab from dashboard
- 2) Coordinator downloads the syllabus of course
- 3) Coordinator accepts the course
- 4) Coordinator adds the course to accepted courses list
- 5) Coordinator clicks save changes
- 6) Accepted courses list is updated

### **Use Case #16**

Use case: Create new role

Actors: Admin

Entry Conditions: - Logged in as admin

Exit Conditions: - Roles list is updated successfully

Flow of Events:

- 1) Admin selects roles page from dashboard
- 2) Admin clicks create
- 3) Admin enters role name and permissions
- 4) Admin clicks save
- 5) Roles list is updated

### **Use Case #17**

Use case: Assign / change roles of user

Actors: Admin

- Entry Conditions: - Logged in as admin
- Exit Conditions: - Role of user updated successfully
- Flow of Events:
- 1) Admin selects user's page from dashboard
  - 2) Admin clicks edit user
  - 3) Admin changes the role attribute of user
  - 4) Admin clicks save changes
  - 5) Role of user is updated

### 4.1.1 Use-Case Model

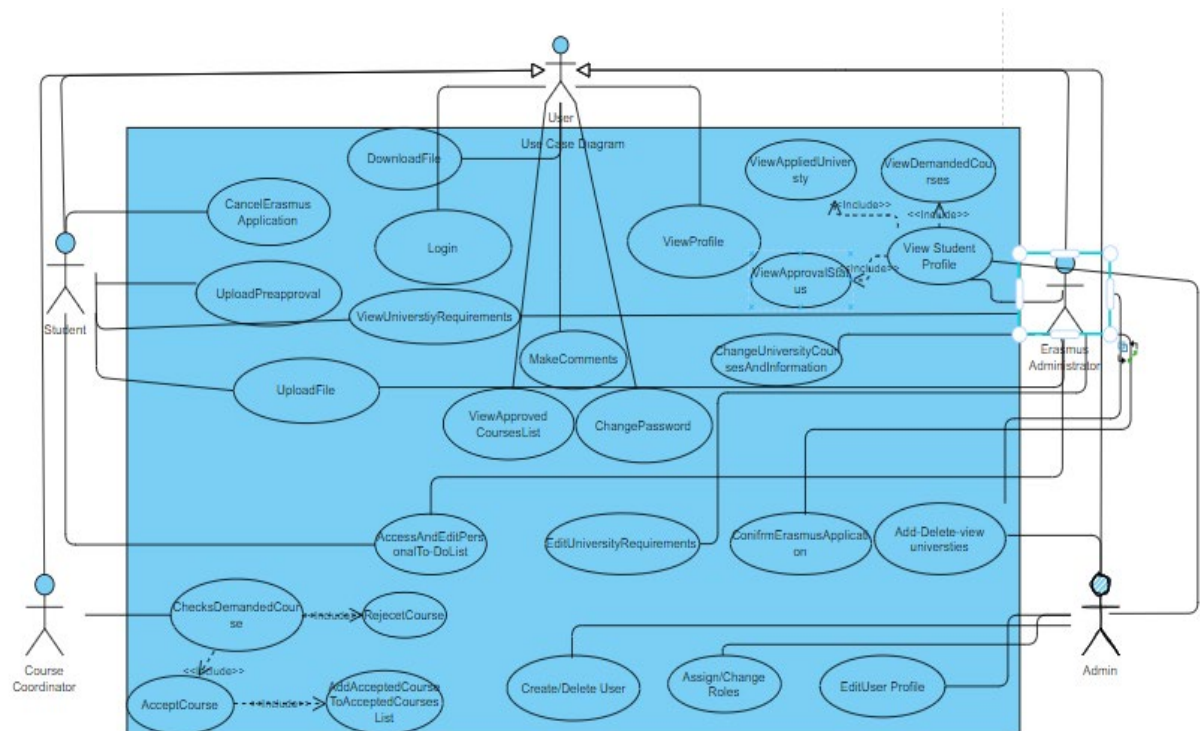


Figure 1 General Use-Case Model

## 4.1.2 Object and Class Model

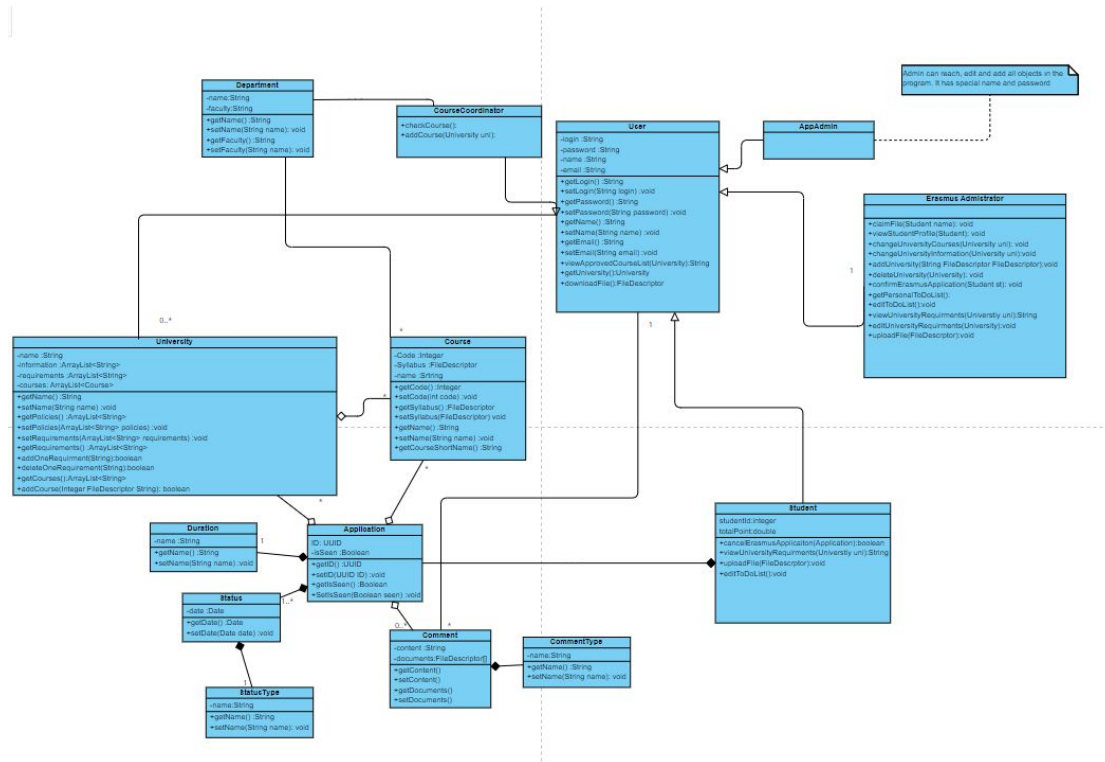


Figure 2 Class Diagram

## 4.2 Dynamic Models

### 4.2.1 Sequence Diagrams

#### Scenario 1: Erasmus administrator confirms student application

Erasmus administrator wants to confirm a student's application. He selects the student from the student list option on his dashboard. Request is made to the database to get the student's application approval status and pre approval status of his courses. If eligible, the student's application is confirmed, the status of his application is updated and he is notified.

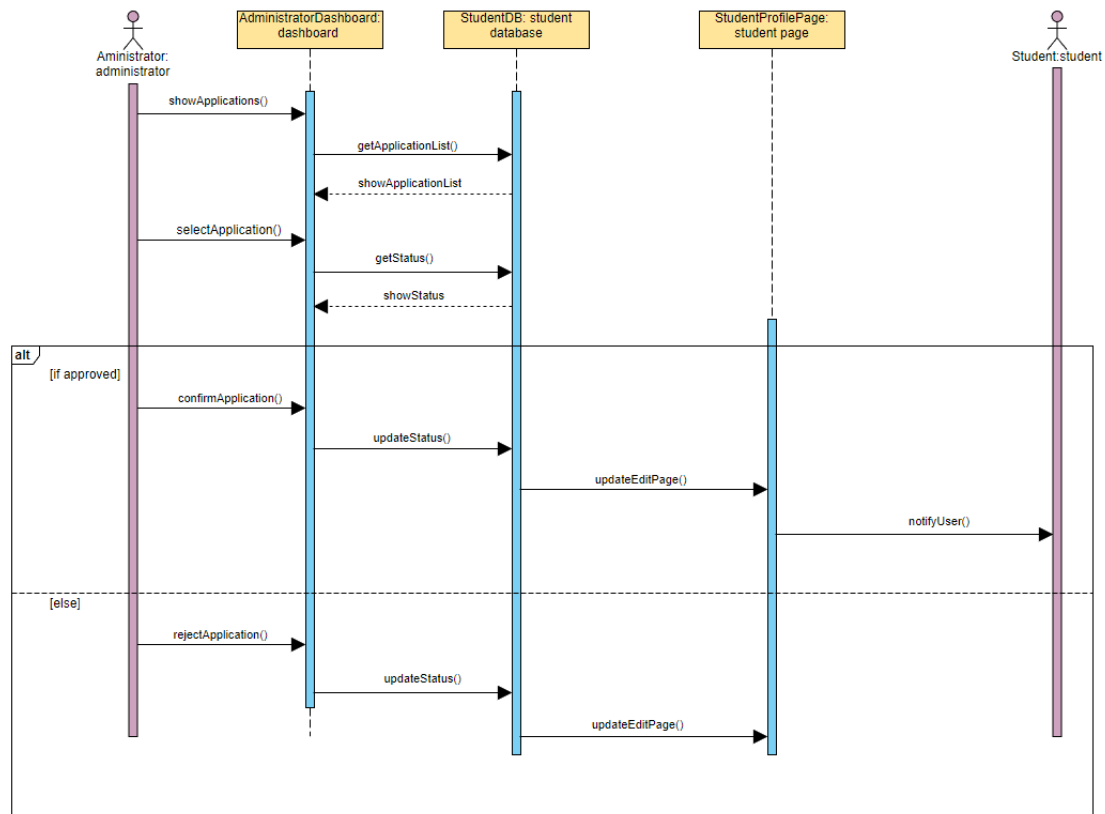


Figure 3 Sequence Diagram for Confirming Erasmus Application

## Scenario 2: Course pre approval syllabus upload

A student wants to pre approve courses that are not in the previously approved courses list. He gathers the syllabuses for the courses and wants to submit them to the coordinator. The Edit Application page has an upload files button that allows students to upload files through the Upload File Page. After the uploaded files are stored, the coordinator is notified that there is a new submission. Once the comments are available by the coordinator, the student is notified.



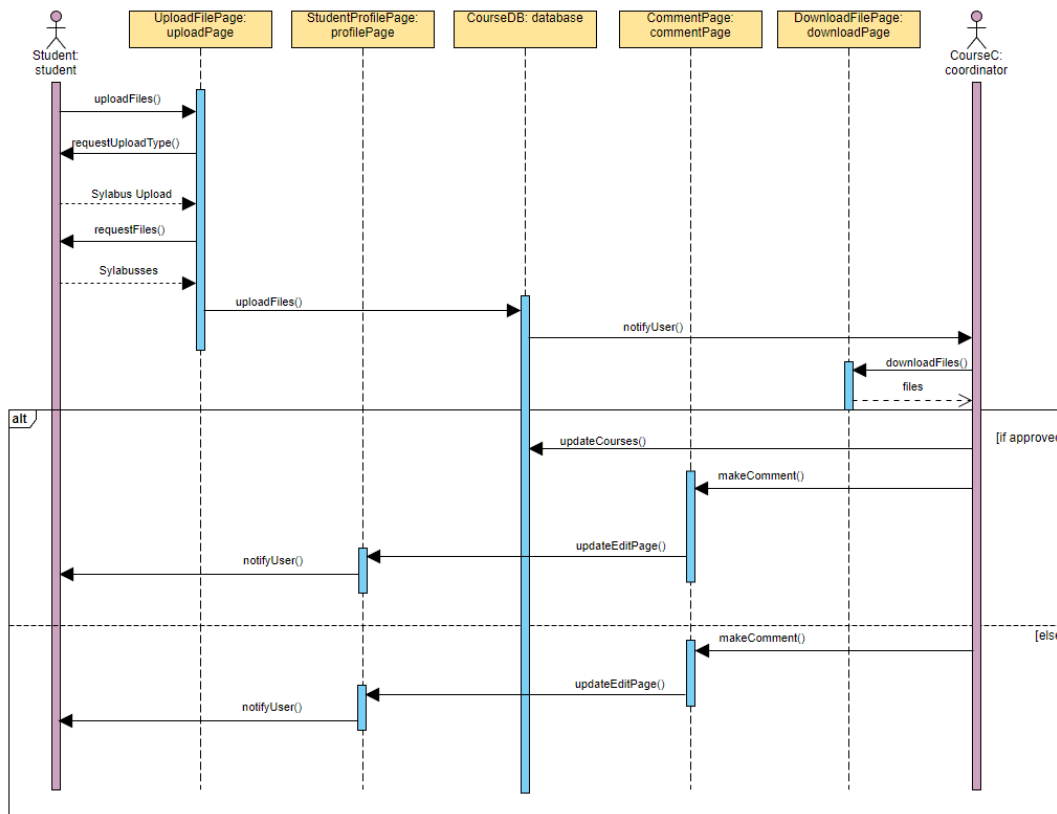


Figure 4 Sequence Diagram for Syllabus Upload

**Scenario 3:** Admin wants to create a new user and set his role, a coordinator for instance. He selects the add user feature and selects coordinator as the role. A new coordinator object is created. Admin sets the data of the coordinator such as name, role, and email. New coordinator is initialized and stored in the database.

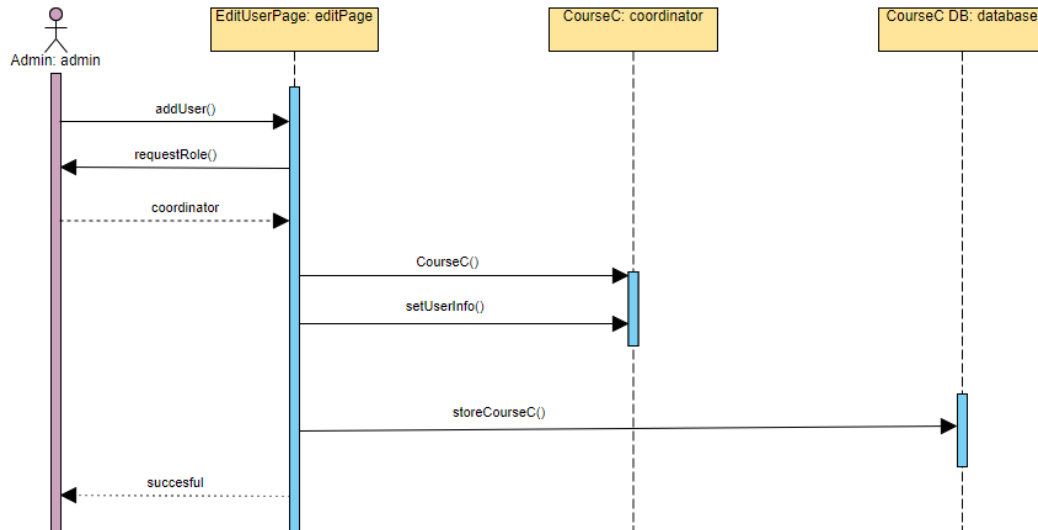


Figure 5 Sequence Diagram for Creating New User and Set His Role

**Scenario 4:** A user wants to log in. He enters his credentials. Application verifies the login information. If login cannot be verified, an error message is displayed and the text fields are cleared for another attempt. If login is verified, a successful login message is displayed and the user is directed to the homepage according to his role. For example a student is directed to the StudentMainPage and an admin is directed to AdminMainPage.

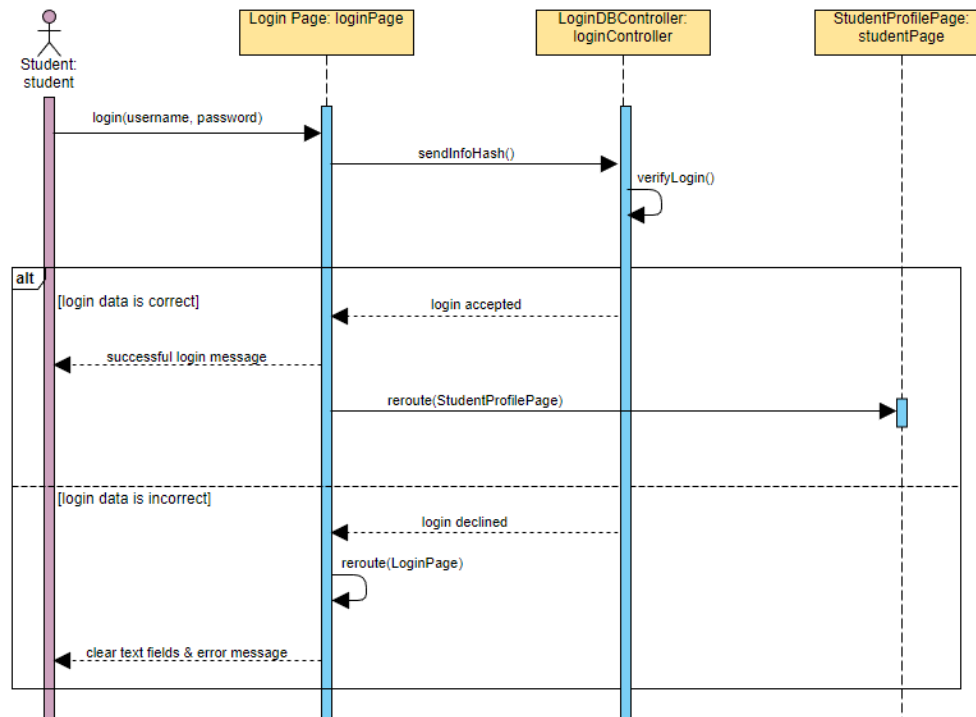


Figure 6 Sequence Diagram for Login

## 4.2.2 State Diagrams

### 1) Student State Diagram

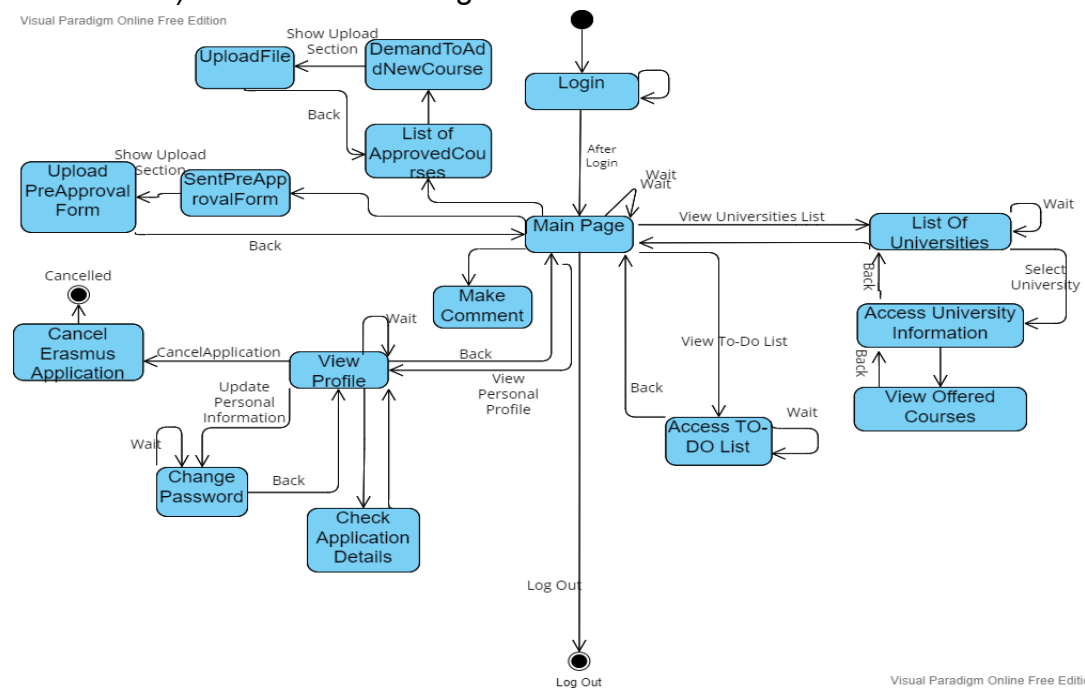


Figure 7 Student State Diagram

## 2) Administrator State Diagram

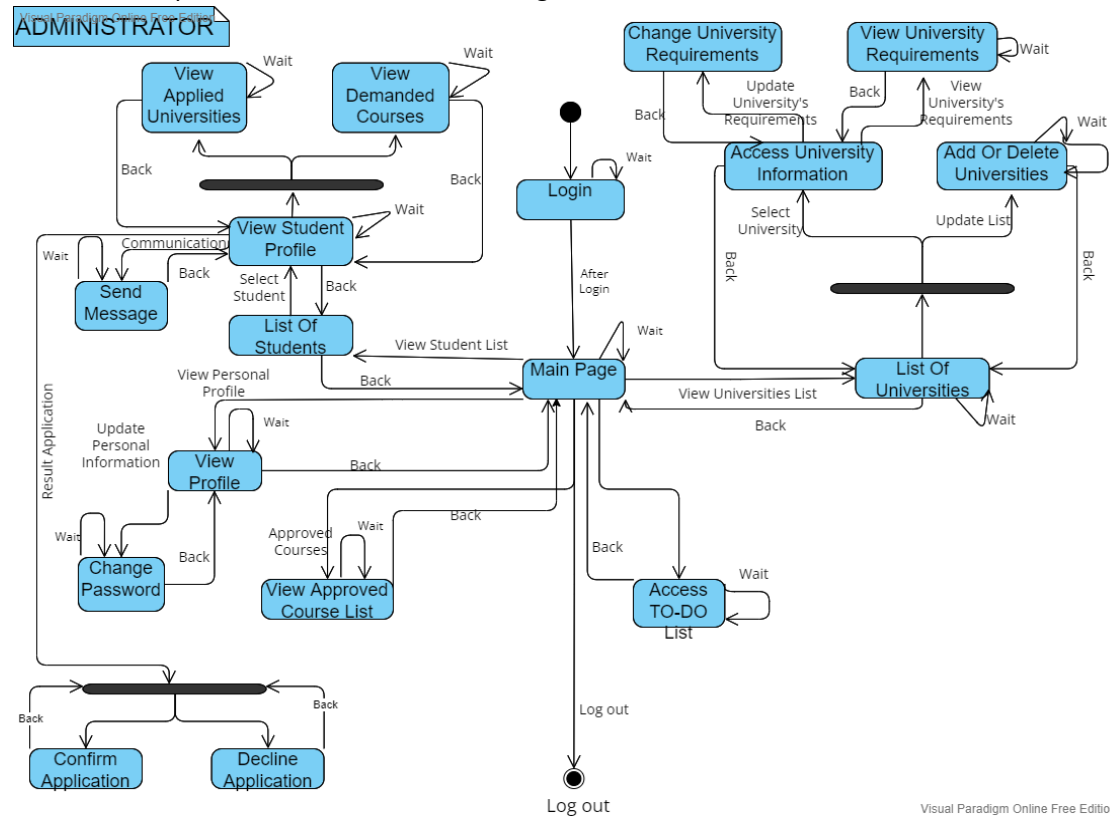
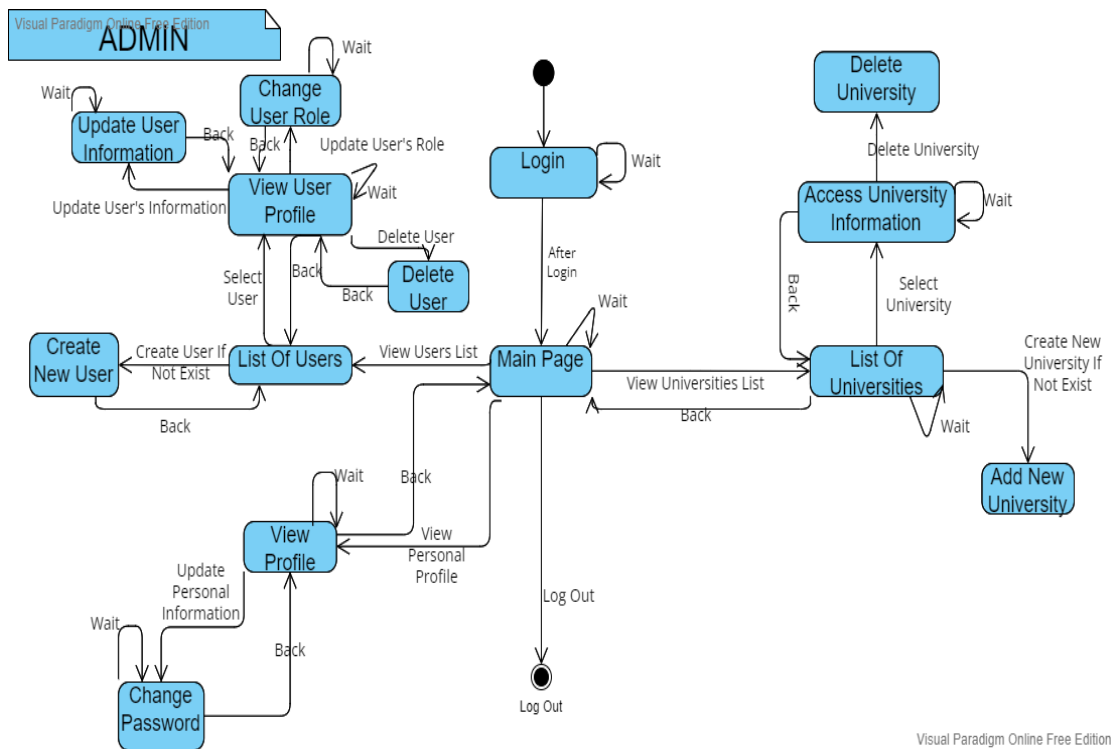
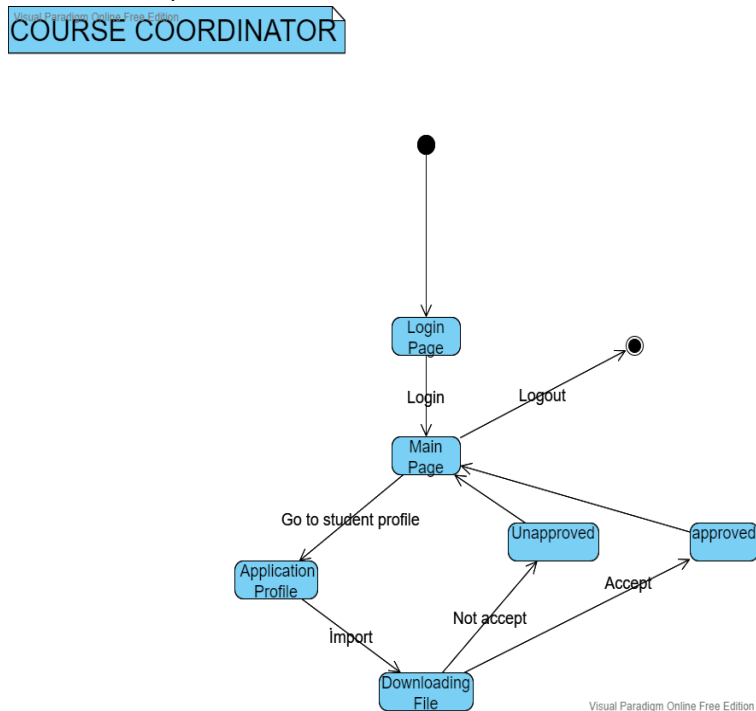


Figure 8 Administrator State Diagram

## 3) Admin State Diagram



#### 4) Course Coordinator



### 4.2.3 Activity Diagrams

#### 1) Application Process Activity Diagram

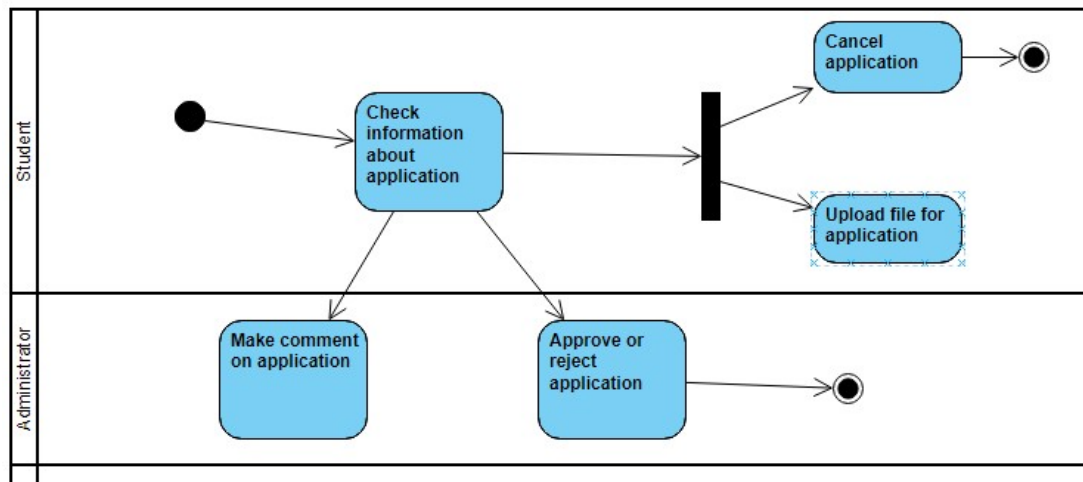


Figure 11 Application Process Activity Diagram

#### 2) Course Process Activity Diagram

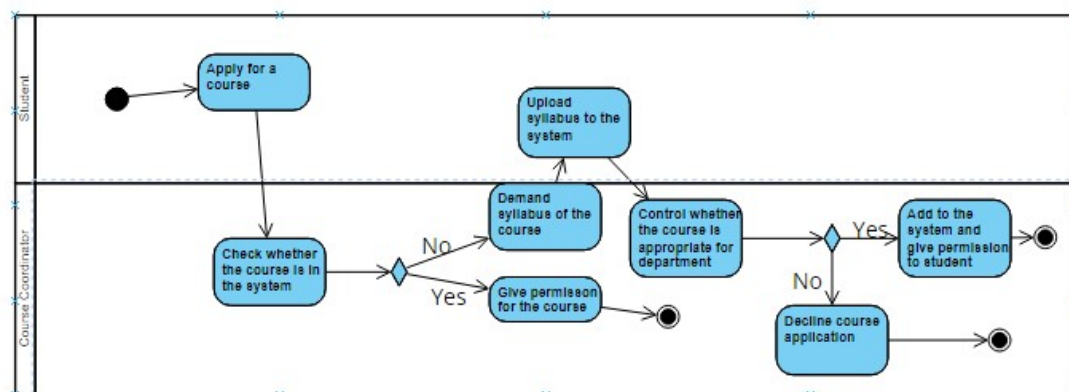


Figure 12 Course Process Activity Diagram

## 5 User Interface

### Login Page

# ErasmusAGA Login

Username:

22211122

Password:

\*\*\*\*\*

Login

Figure 13 Login Page

### Admin Main Page

PROFILE	
APPLICATIONS	
IMPORT EXCEL	
UNIVERSITIES	
ADD UNIVERSITIES	
ROLES	
USERS	

Figure 14 Admin Main Page

## Admin Edit Application Screen

PROFILE	
APPLICATIONS	
IMPORT EXCEL	
UNIVERSITIES	
ADD UNIVERSITIES	
ROLES	
USERS	

ID

2186554

Student Name

mehmet

Student ID

8754998

Total Point

98,65

Duration

spring

Status

approved

University Name	Quota
cambridge	10
munch	2
bilkent	5

Accept

Decline

### Comments

Example comment

Comment Type: Course request

Accept

Decline

Example comment

Select Type

Upload file

Save Comment

Figure 15 Admin Edit Application Page

## Admin Profile

PROFILE	
APPLICATIONS	
IMPORT EXCEL	
UNIVERSITIES	
ADD UNIVERSITIES	
ROLES	
USERS	

Name

mehmet

Password

\*\*\*\*\*

Mail

mehmet@bilkent.edu.tr

SAVE CHANGES

Figure 16 Admin Profile Page



## Admin Applications

[illegible]

*Figure 17 Admin Applications Page*

## Admin Roles Page

IMPORT EXCEL

UNIVERSITIES

ADD UNIVERSITIES

ROLES

USERS

Create

Edit

Remove

Name	Description
Example Name 1	Description
Example Name 2	Description
Example Name 3	Description
Example Name 4	Description
Example Name 5	Description
Example Name 6	Description
Example Name 7	Description
Example Name 8	Description
Example Name 9	Description

Figure 18 Admin Roles Page

## Admin Edit Role Page

IMPORT EXCEL	<b>Roles</b> Name <input type="text"/> Role <input type="text"/> <table><thead><tr><th>Target</th><th>Permission</th></tr></thead><tbody><tr><td>▶ Main Menu</td><td></td></tr><tr><td>▶ All Screens</td><td></td></tr><tr><td></td><td></td></tr><tr><td></td><td></td></tr><tr><td></td><td></td></tr></tbody></table>	Target	Permission	▶ Main Menu		▶ All Screens							
Target		Permission											
▶ Main Menu													
▶ All Screens													
UNIVERSITIES													
ADD UNIVERSITIES													
ROLES													
USERS													

Figure 19 Admin Edit Role Page

## Admin Users Page

IMPORT EXCEL	<input type="button" value="Create"/>	<input type="button" value="Edit"/>	<input type="button" value="Remove"/>																																																																
UNIVERSITIES	<table><thead><tr><th>Login</th><th>Name</th><th>E-mail</th><th>Role</th></tr><tr><th>username</th><th>name-surname</th><th>e-mail</th><th>Student</th></tr></thead><tbody><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr></tbody></table>			Login	Name	E-mail	Role	username	name-surname	e-mail	Student																																																								
Login				Name	E-mail	Role																																																													
username				name-surname	e-mail	Student																																																													
ADD UNIVERSITIES																																																																			
ROLES																																																																			
USERS																																																																			

Figure 20 Admin Users Page

## Admin Edit User Page

IMPORT EXCEL	
UNIVERSITIES	
ADD UNIVERSITIES	
ROLES	
USERS	

Username\*

Password\*

Name\*

Student ID

Department

Email\*

Role\*

Figure 21 Admin Edit User Page

## Erasmus Administrator Main Page

PROFILE	
APPLICATIONS	
IMPORT EXCEL	
UNIVERSITIES	
ADD UNIVERSITIES	

### To do list

Application ID	Application Status	-Duration
-21854566	Approved	06.10.2022
-21854516	Waiting	02.10.2022
-	-	-
-	-	-
-	-	-
-	-	-
-	-	-
-	-	-
-	-	-
-	-	-

Figure 22 Erasmus Administrator Main Page

## Erasmus Administrator Edit Application Screen

PROFILE

APPLICATIONS

IMPORT EXCEL

UNIVERSITIES

ADD UNIVERSITIES

✓

seen

ID2186554

Student Name mehmet

Student ID8754998

Total Point98,65

Durationspring

Statusapproved

University Name	Quota
cambridge	10
munich	2
bilkent	5

Accept

Decline

Comments

Example comment

Comment Type: Course request

Accept

Decline

Example comment

Select Type

Upload file

Save Comment

Figure 23 Erasmus Administrator Edit Application Screen

## Erasmus Administrator Profile

PROFILE

APPLICATIONS

IMPORT EXCEL

UNIVERSITIES

ADD UNIVERSITIES

Name

Password

Mail

mehmet

\*\*\*\*\*

mehmet@bilkent.edu.tr

SAVE CHANGES

Figure 24 Erasmus Administrator Profile Page

[illegible]

## Erasmus Administrator List Courses Screen

[illegible]

37

# Erasmus Administrator Excel Import Screen

PROFILE

APPLICATIONS

IMPORT EXCEL

UNIVERSITIES

ADD UNIVERSITIES

ROLES

USERS

## Import Excel File

Please enter the local path of excel file:

File Pathway

Import File

Figure 27 Erasmus Administrator Excel Import Page

# Erasmus Administrator Universities Screen

PROFILE

APPLICATIONS

IMPORT EXCEL

UNIVERSITIES

ADD UNIVERSITIES

## Universities

University Name	Language Requirements	Offeren Courses	Additional Information
• Munich University	German	-CS 227	None
Cambridge University	English	Eng 102	None
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-

Figure 28 Erasmus Administrator Universities Page

## Course Coordinator Main Page

PROFILE

### To do list

Application ID	Application Status	Duration
21652216	Approved	06.10.2022
21654516	Waiting	02.10.2022
-	-	-
-	-	-
-	-	-
-	-	-
-	-	-
-	-	-
-	-	-

Figure 29 Course Coordinator Main Page

## Course Coordinator Edit Application

PROFILE

ID

216554

Student Name

mehmet

Student ID

8754998

Total Point

98,65

Duration

SPRING

Status

approved

University Name	Quota
cambridge	10
munch	2
bilkent	5

### Comments

Example comment

Comment Type: Course request

Accept

Decline

Example comment

Select Type

Upload file

Save Comment

Figure 30 Course Coordinator Edit Application Page

## Course Coordinator Profile

PROFILE	
Name	mehmet
Password	*****
Mail	mehmet@bilkent.edu.tr
<a href="#">SAVE CHANGES</a>	

Figure 31 Course Coordinator Profile Page

## Student Main Page

PROFILE
APPLICATION STATUS
INFO ABOUT ERASMUS

Figure 32 Student Main Page



## Student Edit Application Screen

PROFILE

APPLICATION STATUS

ID

2186554

Student Name

mehmet

Student ID

8754998

Total Point

98,65

Duration

SPRING

Status

approved

University Name	Quota
cambridge	10
munich	2
bilkent	5

### Comments

Example comment  
Comment Type: Course request

Example comment

Select TypeUpload fileSave Comment

Figure 33 Student Edit Application Page

## Student Profile

PROFILE

APPLICATION STATUS

Name

ID

Password

Mail

GPA

STUDENT ID

Mehmet

218254

\*\*\*\*\*

mehmet.aga@bilkent.edu.tr

3.54

21802665

SAVE CHANGES

Figure 34 Student Profile Page

## Student Approved Course List

PROFILE

APPLICATION STATUS

### Approved course list For Cambridge University

Courses
cs321
~cs433
eng312
ee132
hum113
-
-
-
-

Figure 35 Approved Course List Page

## File Upload Screen

File Upload

Files.txt

⊖ ⬇

Dosya Seç

Figure 36 File upload page

## 6 References

- [1] "Difference between sequence diagram and activity diagram," *GeeksforGeeks*, 06-Mar-2019. [Online]. Available: <https://www.geeksforgeeks.org/difference-between-sequence-diagram-and-activity-diagram/>. [Accessed: 06-Nov-2022].
- [2] "Unified modeling language (UML): Activity Diagrams," *GeeksforGeeks*, 02-May-2022. [Online]. Available: <https://www.geeksforgeeks.org/unified-modeling-language-uml-activity-diagrams/#:~:text=Difference%20between%20a%20Use%20case,not%20how%20it%20is%20done>. [Accessed: 06-Nov-2022].
- [3] L. Lithmee, "Difference between use case diagram and activity diagram," *Compare the Difference Between Similar Terms*, 13-Jun-2018. [Online]. Available: <https://www.differencebetween.com/difference-between-use-case-diagram-and-activity-diagram/#:~:text=What%20is%20the%20Difference%20Between,system%20similar%20to%20a%20flowchart>. [Accessed: 06-Nov-2022].
- [4] "What is Activity Diagram?," *What is activity diagram?* [Online]. Available: <https://www.visual-paradigm.com/guide/uml-unified->

modeling-language/what-is-activity-diagram/. [Accessed: 06-Nov-2022].

[5] W. Limited, "Writing an academic report," *Wordy*, 16-Jan-2018.

[Online]. Available: <https://www.wordy.com/writers-workshop/writing-an-academic-report/>. [Accessed: 06-Nov-2022].