



Bilkent University

Department of Computer Engineering

Senior Design Project

ErasmusAga

Analysis Report

Doruk Altan 21401362 doruk.altan@ug.bilkent.edu.tr

Asım Bilal Ak 21802887 bilal.ak@ug.bilkent.edu.tr

Gökberk Altıparmak 21901798 g.altiparmak@ug.bilkent.edu.tr

Furkan Yıldırım 21902514 f.yildirim@ug.bilkent.edu.tr

Alp Tuğrul Ağçalı 21801799 tugrul.agcali@ug.bilkent.edu.tr

Instructor: Eray Tüzün

Teaching Assistant: İdil Hanhan

Contents

1 Introduction	4
2 Overview	5
2.1 Student	6
2.1.1 Features Of Students	6
2.1.1.1 Profile of Students	6
2.1.1.2 Grades Of Students	6
2.1.1.3 Applications of Students	7
2.1.1.4 Course Selection Of Student's	7
2.1.2 Authority Of Students	8
2.1.2.1 Students will be able to check information about their Application	8
2.1.2.2 Students will be able to cancel their Erasmus application	8
2.1.2.3 Students will be able to upload file to the application	8
2.2 Administrators	9
2.2.1 Administrators will be able to see the student's personal information	9
2.2.2 Administrators will be able to see the applied universities for erasmus by student	9
2.2.3 Administrators will be able to see the the demanded courses to be taken by students at pre-approval stage	10
2.2.4 Administrators will have a to-do list	10
2.2.5 Administrators will confirm the erasmus application at the end of process	10
2.3 Course Coordinators	10
2.3.1 Course Coordinators will be able to check whether demanded course is matched with any course in department	10
2.4 Admins	11
2.4.1 Admins will be able to register new users and deregister existing users	11
2.4.2 Admins will be able to assign roles to the users	11
2.4.3 Admins will be able to add new university or delete the existing university	12
2.4.4 Admins will be able to create new entities or delete the existing entities	12
2.5 Universities	12
2.5.1 Universities will have a course offering list that students can view	12

2.5.2 Universities will have an information page that shows additional or different requirements than Bilkent	13
2.5.3 University requirements can be changed	13
3 Proposed System	14
3.1 Functional Requirements	14
3.1.1 Log in	14
3.1.2 Importing data from local excel file	14
3.1.3 Excel file to database process for utility of information	15
3.1.4 Automatically determining universities for students	15
3.1.5 Course approval	16
3.1.6 Uploading pre approval form	16
3.1.7 Checking Language & Semester Constraints	17
3.2 Additional Requirements	17
3.2.1 To-Do List	17
3.2.2 Approval Status For Submissions	18
3.3 Non-functional Requirements	18
3.3.1 Performance	18
3.3.2 Usability	19
3.3.3 Reliability	19
3.3.4 Supportability	20
4 System Models	20
4.1 Scenarios	20
4.1.1 Use-Case Model	25
4.1.2 Object and Class Model	26
4.2 Dynamic Models	26
4.2.1 Sequence Diagrams	26
4.2.2 State Diagrams	31
4.2.3 Activity Diagrams	34
5 User Interface	36
6 References	48

Analysis Report

Project name: Easy Erasmus

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 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 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 administrators can control this section, but cannot edit it. If there is an error in the imported information, it is forwarded to the

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). The state property will be created as a separate class in order to track the state changes using the creation date of the objects. This separate State class will have a property called stateType which has a property name that can be accepted, being evaluated, in reserve, and denied. The reason why stateType is a separate class is to easily access all of them with a unique ID using the config file. Finally, Applications will have a feature called the accepting university if it is an accepted application. Thanks to this feature, it will be possible to see which university the student has been accepted to.

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 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 Administrators

In the Erasmus application, 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 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 Administrators will be able to see the student's personal information

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 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 Administrators will be able to see the the demanded 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 Administrators will have a to-do list

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

2.2.5 Administrators will confirm the erasmus application at the end of process

When all of the requirements are fulfilled, 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 student, 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, 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 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 can change its policies, alter the courses and acceptance requirements at any time. 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 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.

Administrators and coordinators have password and user name 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. Finally we store data in our database to record information of students.

3.1.3 Excel file to database process for utility of information

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. To avoid compilation all applications are given unique IDs by using life cycle callbacks. 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 administrators with their information as a list. By only double clicking the students name, 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 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 coordinator or Bilkent instructor 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 Faculty Administration Board 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 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 Performance

In order to increase performance in the web application, we preferred that the screens are refreshed when switching from one screen to the other and clicking the refresh button above, instead of periodically refreshing the screens. In addition to these, when we use datagrid, for example, to sort existing applications or to sort existing universities, we will ensure that data is transferred in small groups. While doing this, we will leave the size of the groups to the user and ensure that a maximum of fifty pieces of data are given at the same time. Users will have 5, 10, 25, 50 options.

3.3.2 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, 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 administrator both of them will know what to do for requirements of that erasmus. For data grids, 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.

3.3.3 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.

3.3.4 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: Administrator views student information

Actor: Administrator

Pre-Conditions: - Logged in as administrator
 - Searched student must be a valid, active student

Post-Conditions: - Administrator has access to student information

Flow of Events: 1) Administrator logs in

2) Administrator selects student

3) Administrator accesses student information

Use Case #2

Use case: Administrator views applied universities

Actor: Administrator

Pre-Conditions: - Logged in as administrator

Post-Conditions: - Administrator has access to applied universities list

Flow of Events: 1) Administrator logs in

2) Administrator selects applied universities list

3) Administrator accesses applied universities list

Use Case #3

Use case: Administrator views student approval status

Actor: Administrator

Pre-Conditions: - Logged in as administrator

- Searched student must be a valid, active student

Post-Conditions: - Administrator has access to student approval status

Flow of Events: 1) Administrator logs in

2) Administrator selects student

3) Administrator accesses student approval status

Use Case #4

Use case: Administrator confirms student application

Actor: Administrator

Pre-Conditions:

- Logged in as administrator
- Searched student must be a valid, active student
- Student has pre approved courses
- Student application has approved status

Post-Conditions: - Administrator confirms student application

Flow of Events:

- 1) Administrator logs in
- 2) Administrator selects student
- 3) Administrator checks application status
- 4) Administrator confirms student application
- 5) Student is notified

Use Case #5

Use case: Student applies to universities

Actor: Student

Pre-Conditions:

- Student is eligible for Erasmus
- Student fulfills language and semester requirements

Post-Conditions: - Student applies to universities

Flow of Events: 1) Student selects 5 universities

- 2) Student submits related forms
- 3) Coordinators complete placements
- 4) Student is placed in university or waiting list
- 5) Student is notified

Use Case #6

Use case: Student submits courses for pre approval

Actor: Student

Pre-Conditions:

- Student applied to an institution
- Selected courses are not among previously accepted courses
- Selected courses are similar to must courses here

Post-Conditions: - Course approval status is updated

Flow of Events:

- 1) Student submits syllabuses of the courses
- 2) Coordinator confirms courses
- 3) Student submits pre approval form
- 4) Faculty Administration Committee pre approves or rejects courses
- 5) Student submits the learning agreement for coordinator approval
- 6) Coordinator approves learning agreement

7) Student is notified

Use Case #7

Use case: Student cancels application

Actor: Student

Pre-Conditions: - Student has a placement

Post-Conditions: - Student no longer has a placement

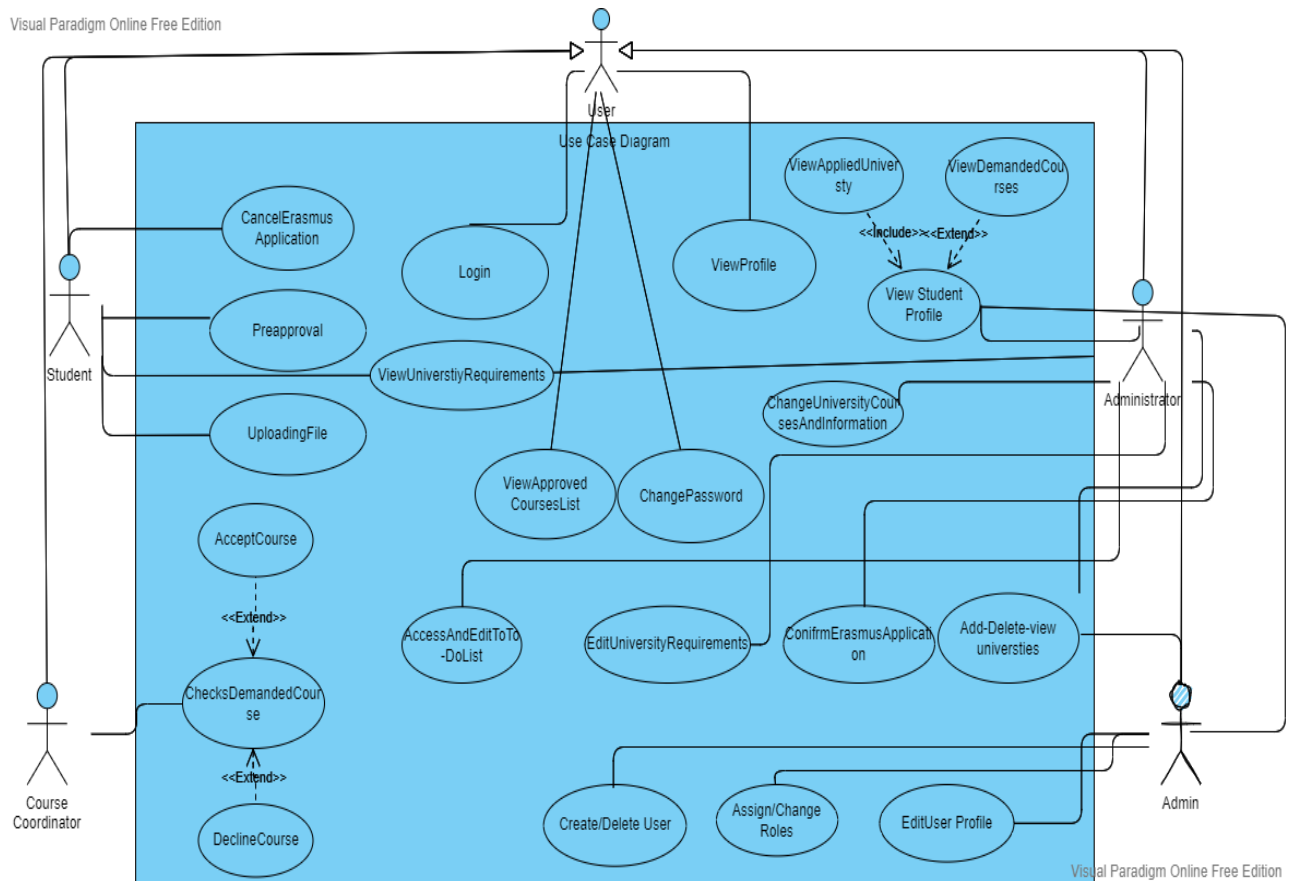
Flow of Events: 1) Student chooses to cancel application

2) Student placement is undone

3) Student is placed in the waiting bin

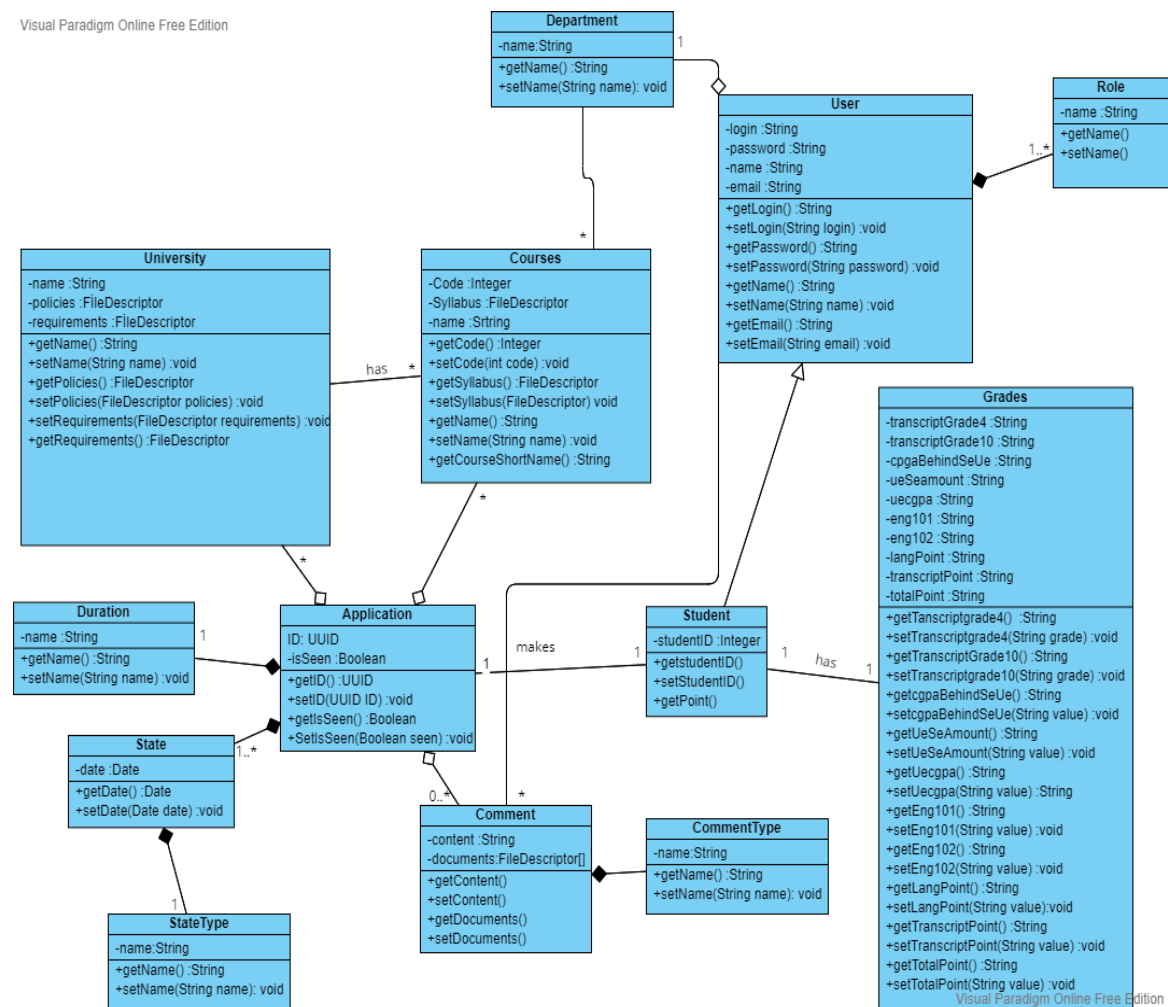
4.1.1 Use-Case Model

Visual Paradigm Online Free Edition



4.1.2 Object and Class Model

Visual Paradigm Online Free Edition



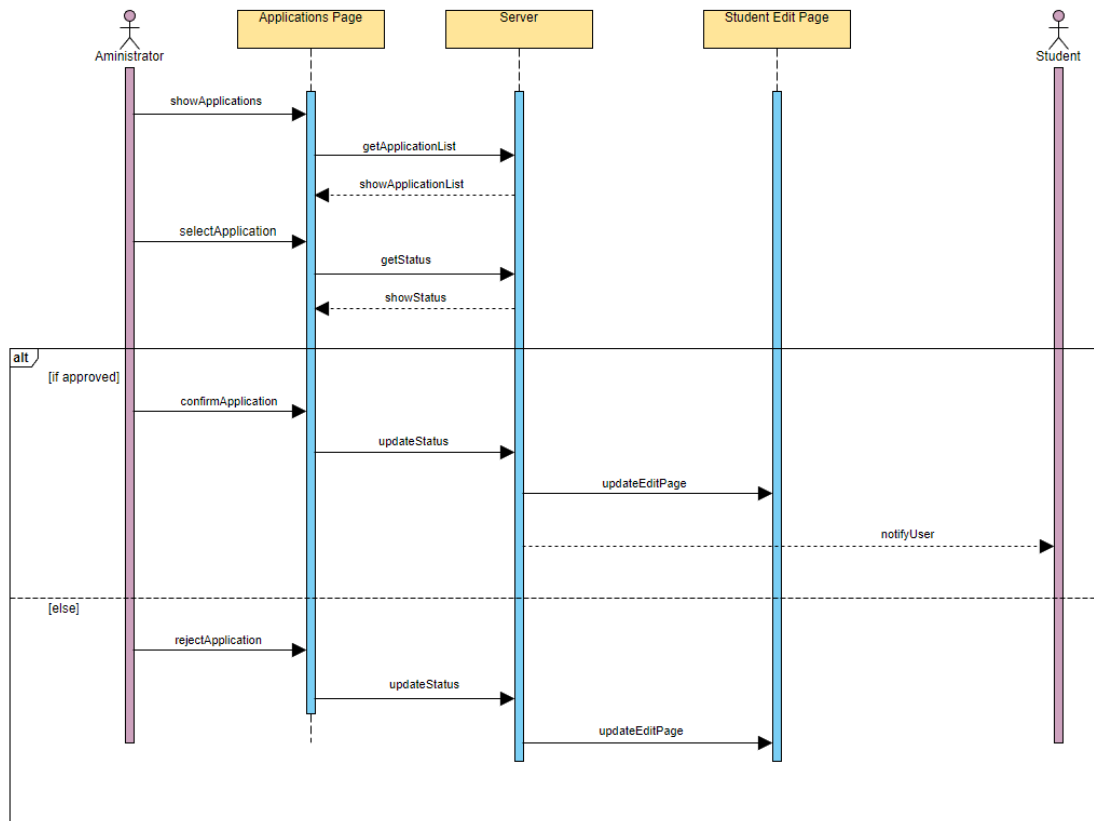
Visual Paradigm Online Free Edition

4.2 Dynamic Models

4.2.1 Sequence Diagrams

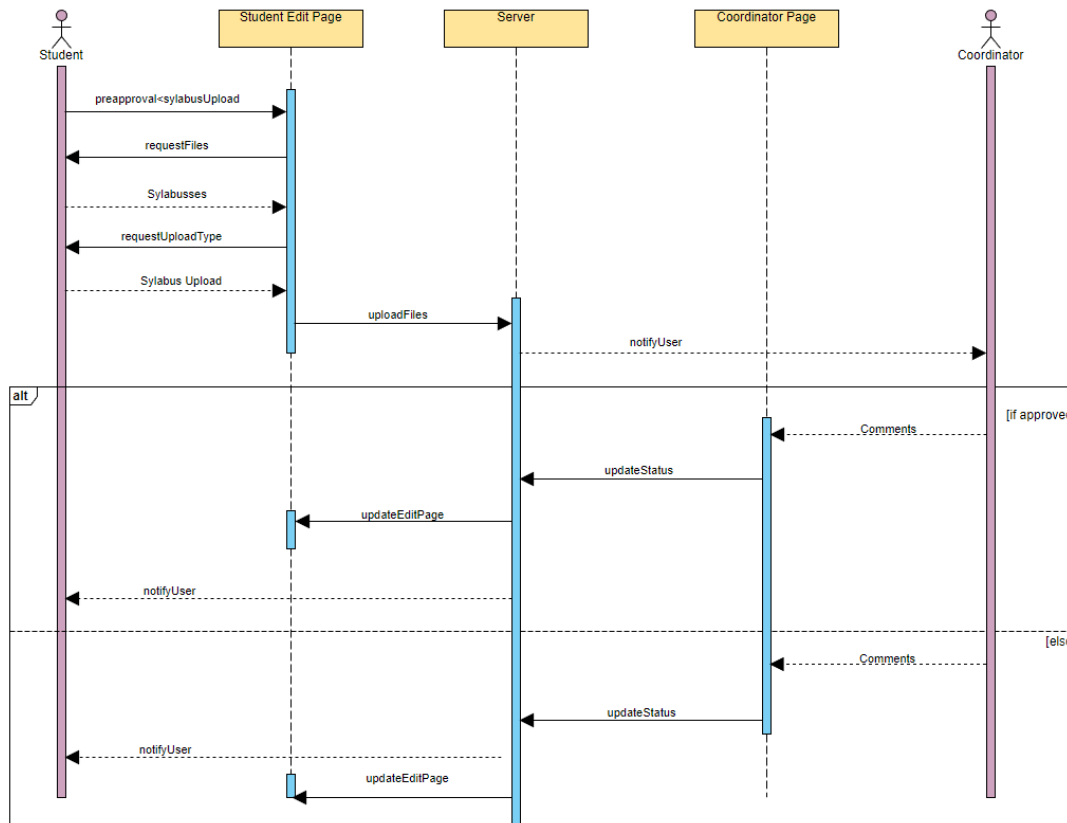
Scenario 1: Administrator confirms student application

Administrator wants to confirm a student's application. He selects the student. Request is made to the server to get the student's approval status from the coordinator and pre approval status of courses. If eligible, the student's application is confirmed.



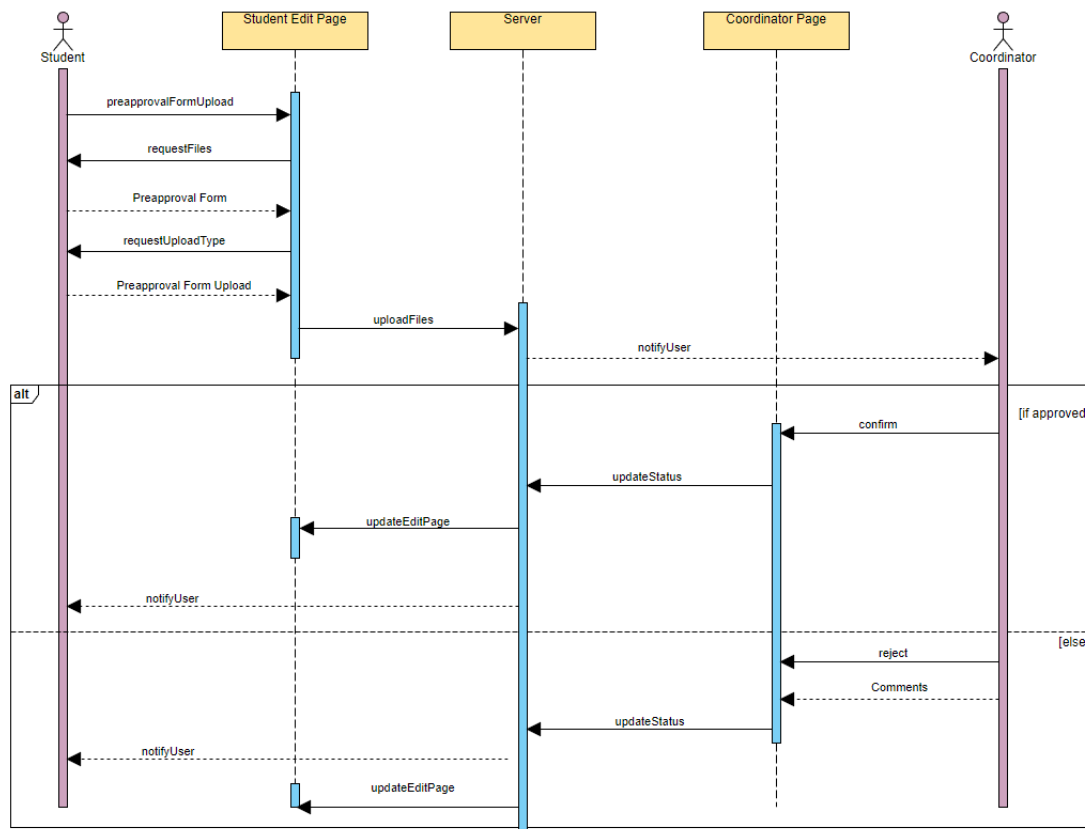
Scenario 2: Course pre approval syllabus upload

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. He uploads the syllabus files. Server informs the coordinator that a submission is made. Once the comments are available by the coordinator, the student is notified.

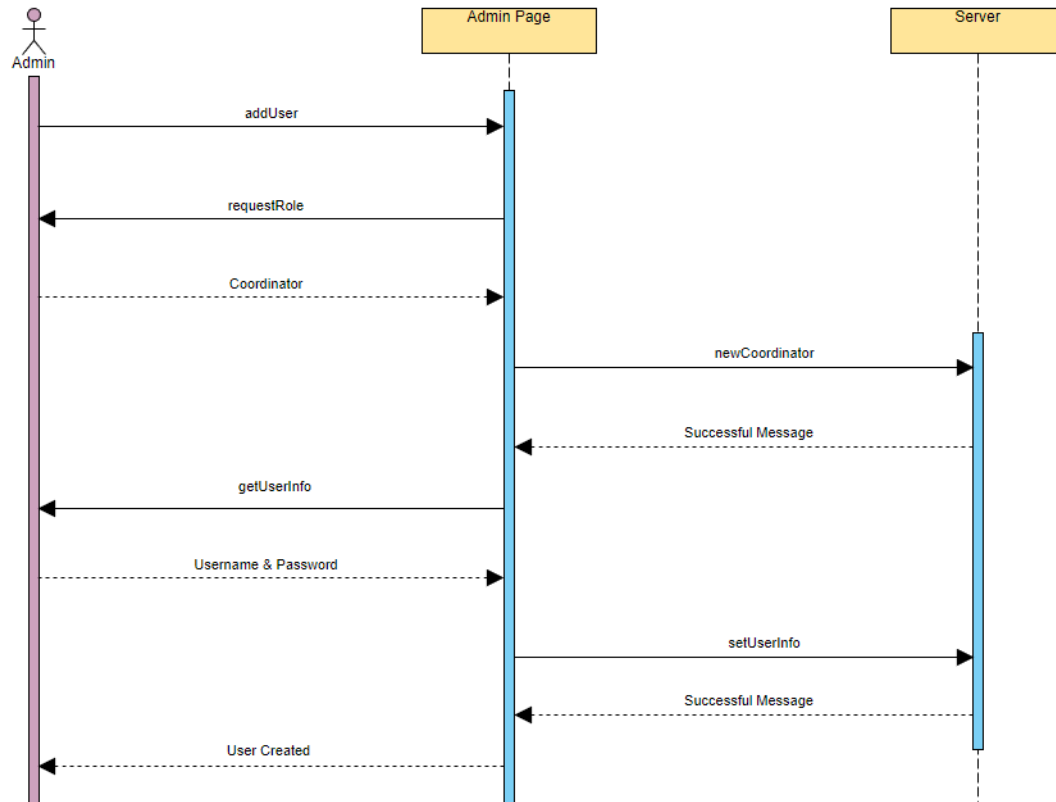


Scenario 3: Pre approval form confirmation

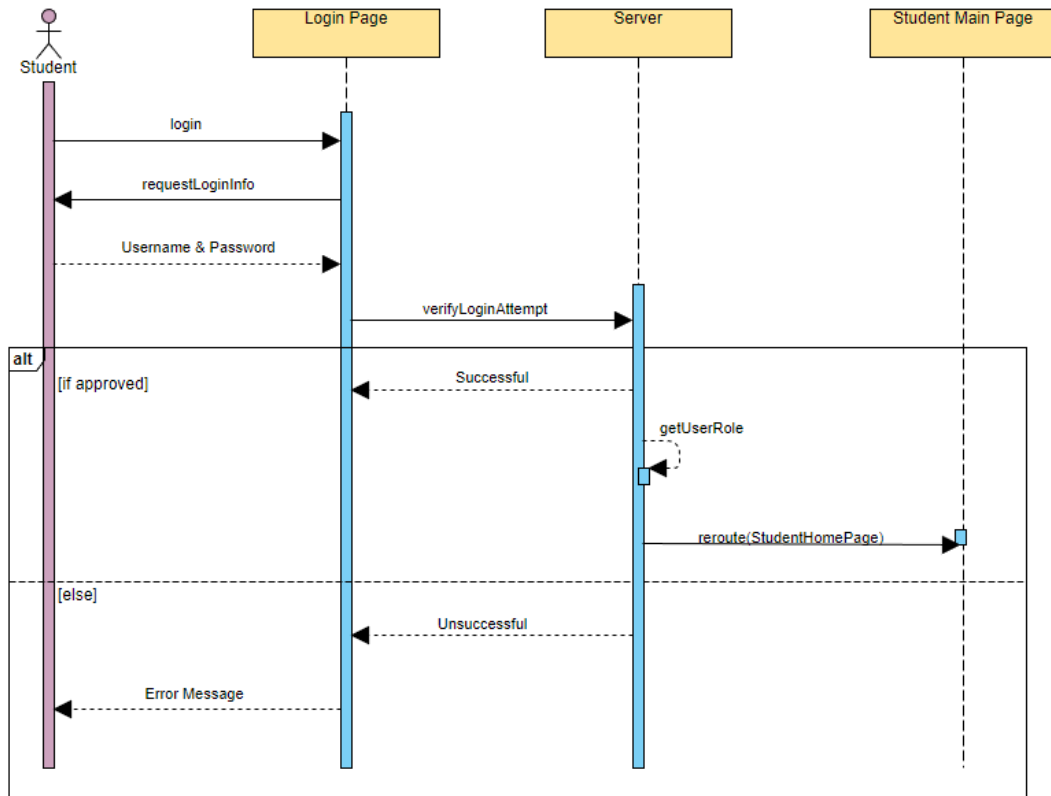
Once a student has selected courses from the previously approved courses list or got confirmation from the coordinator for other courses, he can submit a pre approval form. He selects the upload pre approval form option and uploads the file. Coordinator is notified and if he confirms the upload, the pre approval process is updated as finished, the student's overall application status is updated and the student is notified. If the coordinator does not confirm, he can leave a comment and the student is notified.



Scenario 4: 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. Application creates an empty coordinator object. Application requests the username and password for the new user. Admin sets the information. New coordinator object is initialized.

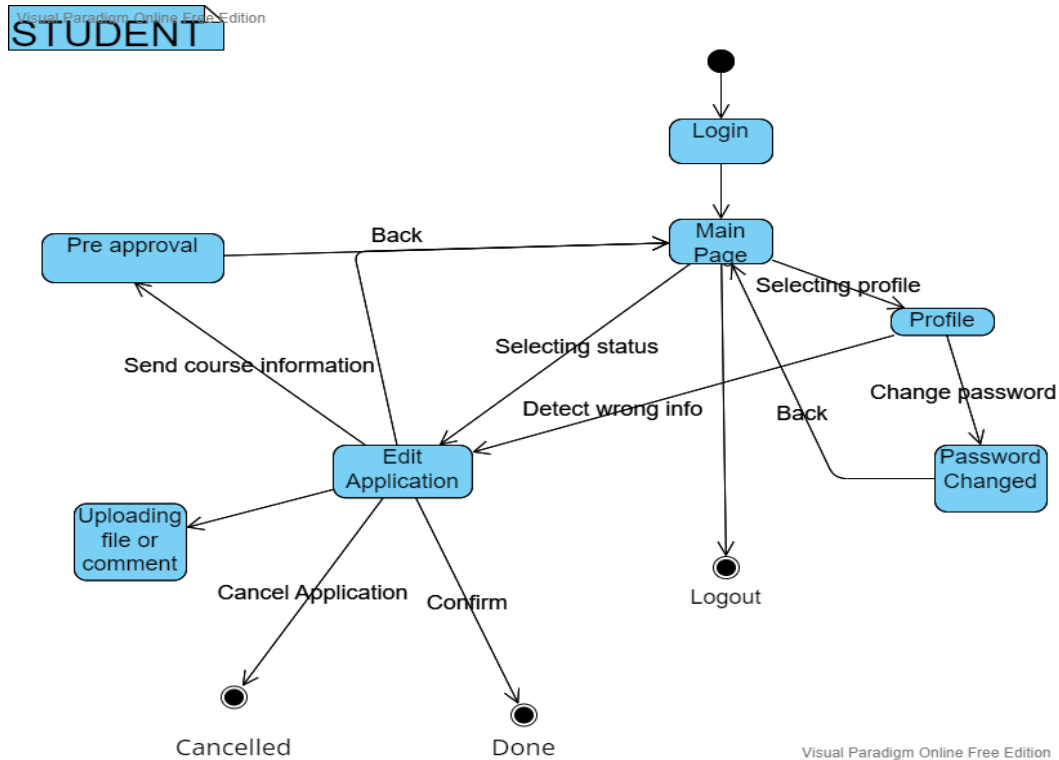


Scenario 5: A user wants to log in. He enters his credentials. Application verifies the login information. If login cannot be verified, an error message is displayed. If login is verified, the user is directed to the homepage according to his role. For example a student is directed to the StudentMainPage and an admin is directed to AdminMainPage.



4.2.2 State Diagrams

1) Student State Diagram



```

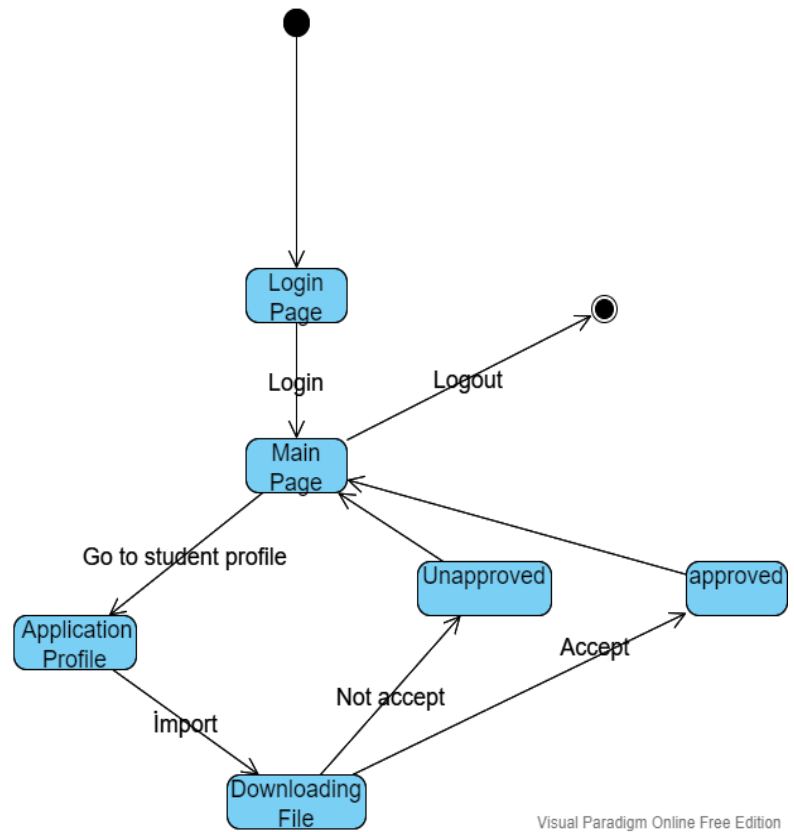
    graph TD
        Start(( )) --> Login[Login]
        Login -- "Wait" --> Login
        Login -- "After Login" --> MainPage[Main Page]
        MainPage -- "Wait" --> MainPage
        MainPage -- "Log out" --> Logout(( ))
        
        Login --> ChangeReq[Change University Requirements]
        ChangeReq -- "Update University's Requirements" --> AccessInfo[Access University Information]
        AccessInfo -- "Back" --> ChangeReq
        AccessInfo -- "Select University" --> AddDel[Add Or Delete Universities]
        AddDel -- "Update List" --> ListUniv[List Of Universities]
        ListUniv -- "Back" --> AddDel
        ListUniv -- "Wait" --> ListUniv
        ListUniv -- "View Universities List" --> MainPage
        AddDel -- "Back" --> ListUniv
        AddDel -- "Wait" --> AddDel
        AddDel -- "View University's Requirements" --> ViewReq[View University Requirements]
        ViewReq -- "Back" --> AddDel
        ViewReq -- "Wait" --> ViewReq
        ViewReq -- "Update University's Requirements" --> ChangeReq
        
        Login --> ViewDem[View Demanded Courses]
        ViewDem -- "Back" --> Login
        ViewDem -- "Wait" --> ViewDem
        ViewDem -- "View Student Profile" --> ViewStuProf[View Student Profile]
        ViewStuProf -- "Back" --> ViewDem
        ViewStuProf -- "Wait" --> ViewStuProf
        ViewStuProf -- "Communication" --> SendMsg[Send Message]
        SendMsg -- "Back" --> ViewStuProf
        SendMsg -- "Wait" --> SendMsg
        SendMsg -- "Result Application" --> ConfApp[Confirm Application]
        SendMsg -- "Result Application" --> DeclApp[Decline Application]
        ConfApp -- "Back" --> SendMsg
        DeclApp -- "Back" --> SendMsg
        
        Login --> ViewApp[View Applied Universities]
        ViewApp -- "Back" --> Login
        ViewApp -- "Wait" --> ViewApp
        ViewApp -- "View Student Profile" --> ViewStuProf
        ViewStuProf -- "Back" --> ViewApp
        ViewStuProf -- "Wait" --> ViewStuProf
        ViewStuProf -- "Select Student" --> ListStu[List Of Students]
        ListStu -- "Back" --> ViewStuProf
        ListStu -- "Wait" --> ListStu
        ListStu -- "View Student List" --> MainPage
        ListStu -- "View Personal Profile" --> ViewProf[View Profile]
        ViewProf -- "Back" --> ListStu
        ViewProf -- "Wait" --> ViewProf
        ViewProf -- "Update Personal Information" --> ChangePass[Change Password]
        ChangePass -- "Back" --> ViewProf
        ChangePass -- "Wait" --> ChangePass
        ChangePass -- "Approved Courses" --> ViewAppList[View Approved Course List]
        ViewAppList -- "Back" --> ChangePass
        ViewAppList -- "Wait" --> ViewAppList
        ViewAppList -- "Back" --> MainPage
        ViewAppList -- "Back" --> ViewAppList
    
```

```

    usecaseDiagram
        [*] --> Login
        Login --> MainPage : After Login
        Login --> Login : Wait
        MainPage --> MainPage : Wait
        MainPage --> Logout : Log Out
        MainPage --> ListUsers : View Users List
        MainPage --> ListUnivs : View Universities List
        MainPage --> ViewProfile : View Personal Profile
        ListUsers --> MainPage : Back
        ListUnivs --> MainPage : Back
        ViewProfile --> MainPage : Back
        ListUsers --> CreateNewUser : Create User If Not Exist
        CreateNewUser --> ListUsers : Back
        ListUsers --> ViewUserProfile : Select User
        ViewUserProfile --> ListUsers : Back
        ViewUserProfile --> UpdateUserInfo : Update User's Information
        UpdateUserInfo --> ViewUserProfile : Back
        ViewUserProfile --> ChangeUserRole : Update User's Role
        ChangeUserRole --> ViewUserProfile : Back
        ViewUserProfile --> DeleteUser : Delete User
        DeleteUser --> ViewUserProfile : Back
        ViewUserProfile --> UpdateUserInfo : Update User's Information
        UpdateUserInfo --> ViewUserProfile : Back
        ViewUserProfile --> ChangePassword : Update Personal Information
        ChangePassword --> ViewUserProfile : Back
        ListUnivs --> AddNewUniv : Create New University If Not Exist
        AddNewUniv --> ListUnivs : Back
        ListUnivs --> AccessUnivInfo : Select University
        AccessUnivInfo --> ListUnivs : Back
        AccessUnivInfo --> DeleteUniv : Delete University
        DeleteUniv --> AccessUnivInfo : Back
    
```


4) Course Coordinator

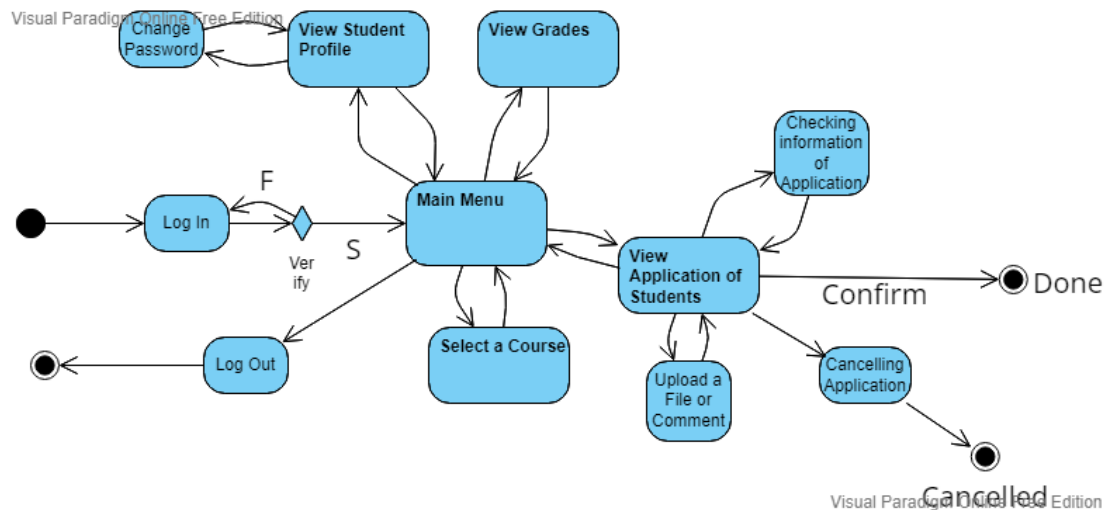
COURSE COORDINATOR



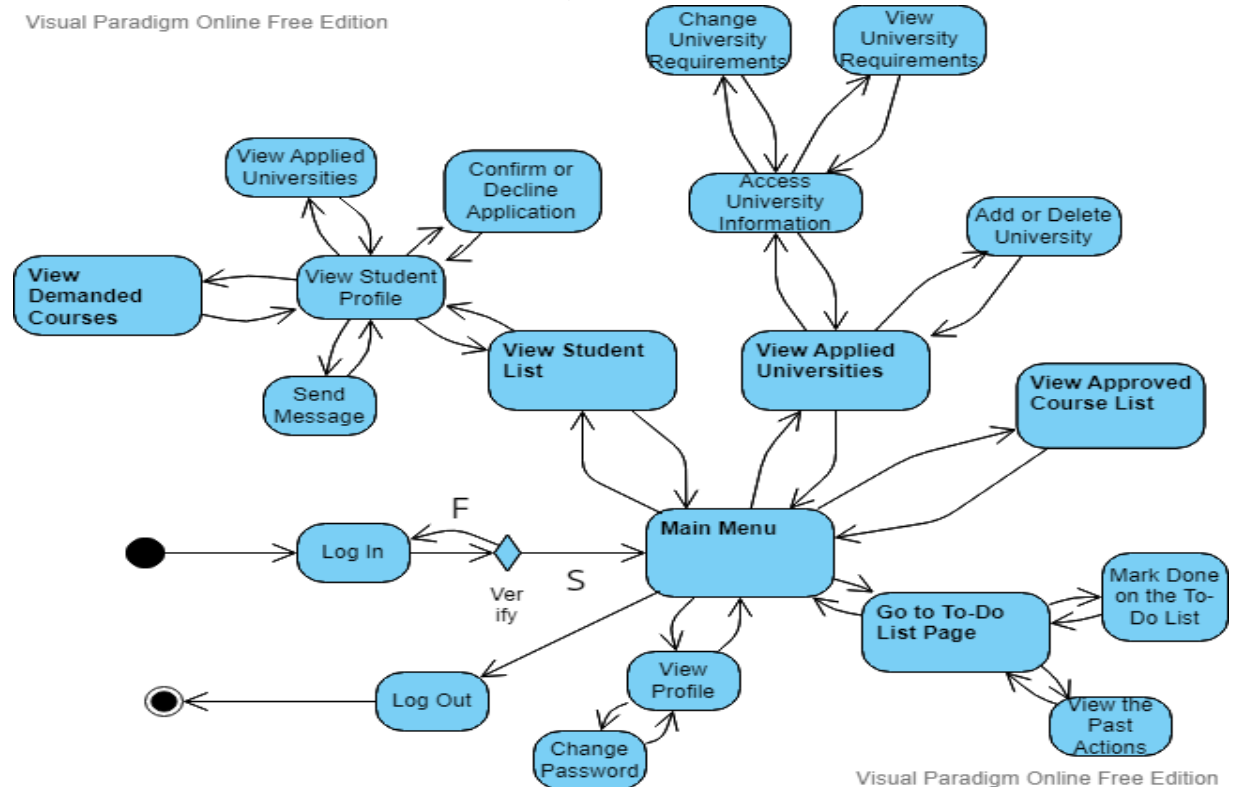
Visual Paradigm Online Free Edition

4.2.3 Activity Diagrams

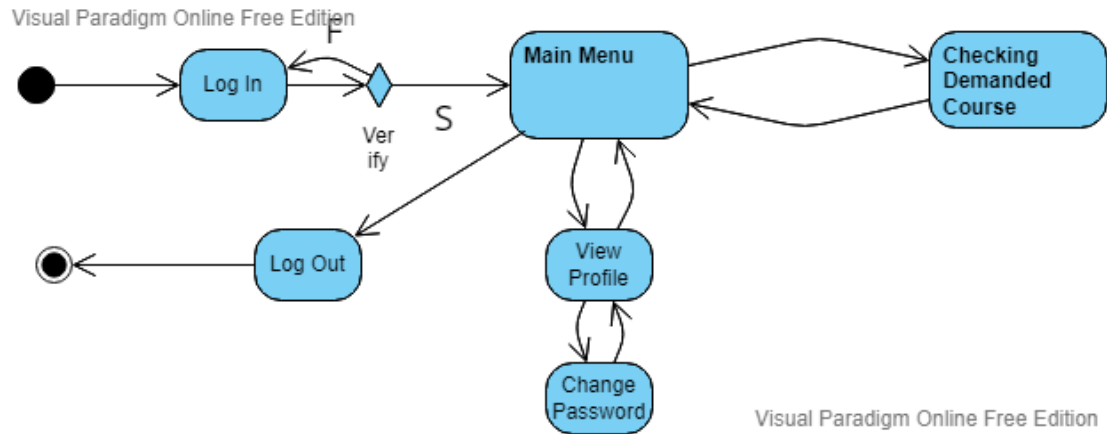
1) Students Activity Diagram



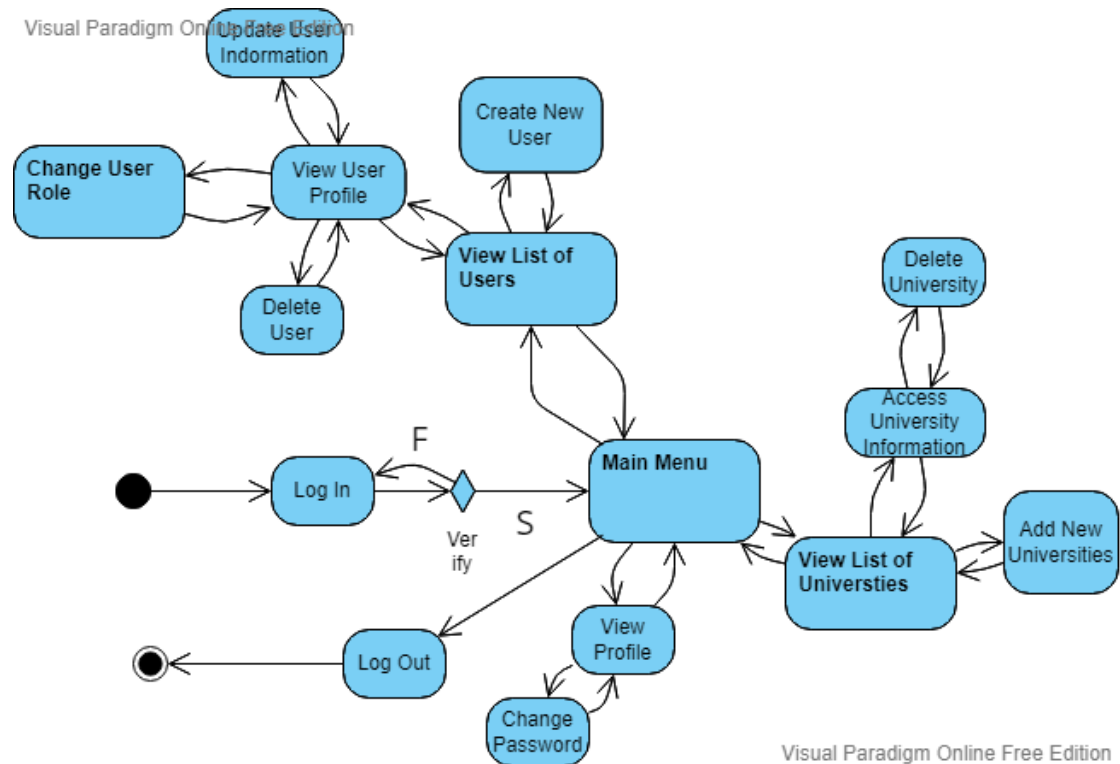
2) Administrator Activity Diagram



3) Course Coordinator Activity Diagram



4) Admin Activity Diagram



5 User Interface

Login Page

Username

login123

Password

asd123

Login

Admin Main Page

PROFILE
APPLICATIONS
IMPORT EXCEL
UNIVERSITIES
ADD UNIVERSITIES
ROLES
USERS

Admin Edit Application Screen

PROFILE

APPLICATIONS

IMPORT EXCEL

UNIVERSITIES

ADD UNIVERSITIES

ROLES

USERS

ID

Student Name

Student ID

Total Point

Duration

State

University Name	Quota

Accept

Decline

Comments

Example comment

Comment Type: Course request

Accept

Decline

Example comment

Select Type

Upload file

Save Comment

Admin Profile

PROFILE

APPLICATIONS

IMPORT EXCEL

UNIVERSITIES

ADD UNIVERSITIES

ROLES

USERS

Name

Password

Mail

SAVE CHANGES

Admin Applications

[illegible]

Admin Roles Page

IMPORT EXCEL

UNIVERSITIES

ADD UNIVERSITIES

ROLES

USERS

CreateEditRemove

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

IMPORT EXCEL

UNIVERSITIES

ADD UNIVERSITIES

ROLES

USERS

Roles

Name

Role

Target	Permission
▶ Main Menu	
▶ All Screens	

[illegible]

Admin Edit User Page

IMPORT EXCEL

UNIVERSITIES

ADD UNIVERSITIES

ROLES

USERS

User

Login

login123

Password

asd123

Name

name123

Student ID

Department

E-mail

email123

Role

Administrator

Administrator Main Page

PROFILE

APPLICATIONS

IMPORT EXCEL

UNIVERSITIES

ADD UNIVERSITIES

To do list

Application ID	Application Status	-Duration
-	-	-
-	-	-
-	-	-
-	-	-
-	-	-
-	-	-
-	-	-
-	-	-
-	-	-

Administrator Edit Application Screen

PROFILE

APPLICATIONS

IMPORT EXCEL

UNIVERSITIES

ADD UNIVERSITIES

☒ seen

ID

Student Name

Student ID

Total Point

Duration

State

University Name	Quota

AcceptDecline

Comments

Example comment

Comment Type: Course request

Example comment

Select TypeUpload fileSave Comment

Administrator Profile

PROFILE

APPLICATIONS

IMPORT EXCEL

UNIVERSITIES

ADD UNIVERSITIES

Name

Password

Mail

SAVE CHANGES

PROFILE
APPLICATIONS
IMPORT EXCEL
UNIVERSITIES
ADD UNIVERSITIES
ROLES
USERS

➤

- IMPORT EXCEL
- UNIVERSITIES
- ADD UNIVERSITIES
- ROLES
- USERS

Remove

[illegible]

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

Administrator Universities Screen

administrator Universities

PROFILE

APPLICATIONS

IMPORT EXCEL

UNIVERSITIES

ADD UNIVERSITIES

Universities

University Name	Language Requirements	Offeren Courses	Additional Information
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-

Course Coordinator Main Page

PROFILE

To do list

Application ID	Application Status	-Duration
-	-	-
-	-	-
-	-	-
-	-	-
-	-	-
-	-	-
-	-	-
-	-	-
-	-	-
-	-	-

Course Coordinator Edit Application Screen

PROFILE

ID

Student Name

Student ID

Total Point

Duration

State

University Name	Quota

Comments

Example comment

Comment Type: Course request

Accept

Decline

Example comment

Select Type

Upload file

Save Comment

Course Coordinator Profile

PROFILE

Name

Password

Mail

SAVE CHANGES

Student Main Page

PROFILE

APPLICATION STATUS

INFO ABOUT ERASMUS

Student Edit Application Screen

PROFILE

APPLICATION STATUS

ID

Student Name

Student ID

Total Point

Duration

State

Universities

University Name	Quota

Comments

Example comment

Comment Type: Course request

Example comment

Select TypeUpload fileSave Comment

Student Profile

PROFILE

APPLICATION STATUS

Name

ID

Password

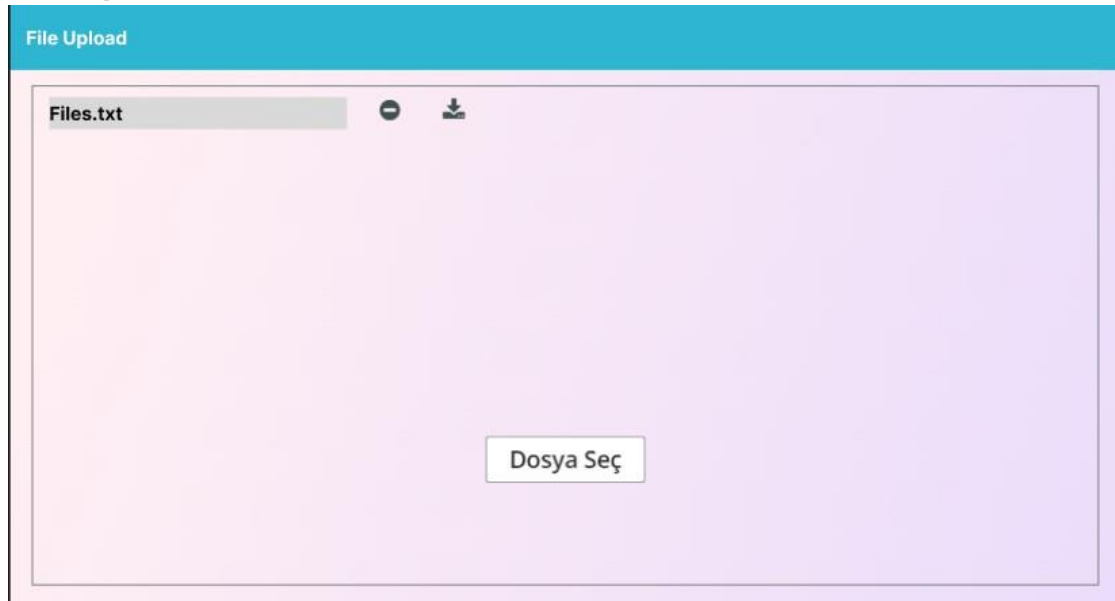
Mail

GPA

STUDENT ID

SAVE CHANGES

File Upload Screen



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].