ÇANKAYA UNIVERSITY FACULTY OF ENGINEERING

COMPUTER ENGINEERING DEPARTMENT

Project Report

CENG 407

Innovative System Design and Development I

P202112

Student Transfer Management System for Engineering Faculty

Cenk ERKILINÇ
201711026
Kubilay AKSEL
201711002
Özlem Nur ERDOĞAN
201711024
Zehra Ece AKDEMİR
201811002

Advisor: Dr. Öğr. Üyesi *Abdül Kadir GÖRÜR*

Contents

Abstract

In Çankaya University, course transfers(or credit transfers) are made for three reasons in accordance with the regulations and directives; Undergraduate transfer, vertical transfer (via national DGS exam), and courses taken and completed at other universities (i.e., during summer school). During this period lots of request forms are signed and send to departments and all forms must be inspected by related instructor. Since all these operations are carried out by single person with in a limited time period, it is error prone . To minimize the mistakes due to nature of the process and reduce the overall processing time, there is a need for a web based course transfer automation system for the faculty which will help transfer coordinators manage course transfer process.

Özet

Üniversitemize yatay veya dikey geçiş yapmak isteyen öğrenciler transcriptleri ile başvurup derslerinin denkliğini ve üniversiteye kabulünü bekliyor. Bu esnada derslerin kontrol edilmesi için koordinatörler, bölüm başkanları ve öğretim görevlileri arasında sürekli bir belge alışverişi oluyor ve en son hali dekana gönderilip onay bekleniyor. Bu işlerin elden yürümesi hem zahmetli hem de dosya kaybı veya teslim edilecek belgelerin unutulması gibi işi aksatabilecek sorunlara açık. Bu nedenle sistemin daha güvenli ve kolay işlemesi açısından transfer işlemlerin bir web sitesi üzerinden yürütüldüğü bir sistem geliştirilecek.

Intoduction

This document is an overview for the web based Student Transfer Management System for Engineering Faculty (STMS) project. The documnet includes information about the literature report which gives perspective about similar projects to the Student Transfer management System for Engineering Faculty. The software requirements specifications document outlines the functions and purpose of the future software product, what the system will do and how it will perform. The Software Design Document does not only define the product functions, user characteristics, constraints, and specific requirements of the system but also serve as a basis for the The document will not only define the product functions, user characteristics, constraints, and specific requirements of the system. The Project Report includes all the theoretical information about the software system

Literarure

Introduction

The course transfer process is carried at each department by transfer coordinators in a laborious and error-prone manner. The process starts at the department upon receiving hard copies of application forms. The coordinator inspects the documents and extracts the list of courses. For each course, he/she prepares a course equivalence opinion request (coer) form. The coer forms have to be signed by the department's head and sent to the related department for the opinion. The related department returns the forms by either accepting or rejecting the decision. Then the coordinator compiles all these required forms and prepares another form where only the accepted courses are listed. This process is done for each student during the undergraduate and vertical transfer periods. The request forms, which are examined and signed by the academic staff, are returned to the relevant departments with the necessary decisions. These operations are open for errors because there is limited time for the transactions done by the department coordinator. Also since these documents are transferred by hand there is a high risk of missing the paper. A course transfer automation system is needed for the faculty to help the transfer coordinators manage the course transfer process in order to minimize the errors arising from the nature of the process and reduce the total processing time. The system that we will develop aims to manage the document processes in an easier, more organized and faster way. In the Student Transfer Management System for Engineering Faculty the process starts with the documents uploads by the coordinator. The coordinator will decide which instructors will inspect the document. To perform the decision the system will send a link to the chosen instructors via email. The instructors can reach the system by clicking on the link. The instructor can reach the system without creating an account easily and safely only by using the sended link. By opening the link there will be a page that contains the transcript of the student, the course schedule of the related department and the course equivalence opinion request (coer) form of the related student. On this page the instructor can decide on the course equivalency and accept or reject the request by choosing the according checkbox. After all course equivalencies are checked by the related instructors the STMS will provide a new course schedule for the student as an output which will be approved by the head of department.

Risk Elements in STMS

Technical-Sided Risks:

- Security
- STMS perception
- System Preference
- Technical Compatibility

Management Sided Risks:

- Organizational Change
- Staff
- Experience
- Project Management
- Communication
- Documentation

Frequent Problems

- Low computer literacy of the staff and fear of computers
- Lack of full support from top management
- Business processes for undefined and/or non-standardized document management.
- Duties and responsibilities are not clearly defined in the STMS configuration process
- Problems with application software, such as not being user-friendly
- Increased workload(To carry out document-related transactions in the physical environment as well as in the electronic environment.)
- Lack of experience (inexperience of the institution personnel and company personnel configuring STMS).

System Frameworks

What is the Microsoft Visual Studio?

Microsoft Visual Studio is an IDE for Windows, an integrated development environment. Visual Studio is a useful program for writing native code and code for websites, web applications, and web services along with managed code for all platforms supported by Microsoft Windows, Windows Mobile, Windows CE, .NET Framework, .NET Compact Framework, and Microsoft Silverlight. What distinguishes this program from other similar programs and the reason why it is preferred by users is that it offers various features such as code editor, debugger, GUI design tool and database schema design option, revision control system. [2]

What is the Angular?

Angular JS is a structural framework for dynamic web applications. Although it allows to use HTML as a template language, one of its biggest advantages is that it is supported by google. For this reason, it is widely used. Angular eliminates most of the code without having to write bulk code, making it an ideal partner to work with almost any server technology found in most browsers. It is completely based on HTML and Javascript. For this reason, there is no need to know and learn any other syntax. It has many different structures such as Data Building, Routing, Templates, which are actively used.[3]

What is the SQL?

SQL is a database system that you can use to easily manage data. The acronym SQL stands for "Structured Query Language". Contrary to popular belief, SQL is not a programming language. As a sub-language, SQL allows you to perform the operations you want in databases.[4]

What is the OCR(Optical Character Recognition)?

These software, called 'Optical Character Recognition' or 'Optical Character Reader', are software developed to transfer the texts in a scanned document, a photograph, a handwritten text to the computer environment. Under normal conditions, graphic files transferred to the computer via a browser are perceived as images by the computer. The texts in this graphic file can be analyzed by 'OCR' software and converted into a text file. Thus, the text will be transferred to the computer without the need to use a keyboard. Data that has been textured with 'OCR' are data that can be edited. Text files take up less space on the computer than image files. Although the error rate of 'OCR' software is not zero, its use has

become quite common today. When it comes to converting photos to text, what is meant is 'OCR' software.[5]

Intuitive Design

An intuitive interface is an interface that works exactly as the user expects it to. For example, when a user sees a button, they assume that if they click that button it will perform a specific action. We don’t have to think about how that button works, we just know how it works. Good intuitive design, then, is user-friendly and offers good UX.[6]

As for website design, the design is intuitive as long as the user can complete the task without interruption. On the other hand, intuitive design changes focus Users' items are not related to their task. [7]

Intuitive User Interface is a user-friendly UI without having to get lost, confused, guessing, experimenting, reading a manual book, or even asking the others.[8]

Similar Systems to our Project
DocuWare Document Management System

As AVANTGARDE SOFTWARE, we start the process by first listening to your needs and priorities. Then, we design features such as document detection, archiving, extraction and processing of metadata, and automation of workflows in accordance with your company's processes DocuWare is a no-code ready-to-use platform. Available as cloud-based, DocuWare can work either as a standalone workflow solution or as an integrated extension to your finance, sales or human resources software.[9]

EBYS Sağlık Bakanlığı

It is a system that aims to transfer documents and information exchange in the institution to an electronic environment and to manage this information instantly over the internet. It also standardizes your internal and external correspondence processes. It is also aimed to minimize the resources spent for correspondence. Prevents or minimizes the problems in the physical environment. Documents in organizations have to be kept and managed in a certain order because they document the activities carried out. The document management discipline is a discipline that ensures the control and arrangement of documents within standard rules throughout all stages from production to archiving in organizations.[10]

en Vision

enVision®, an ECM platform offered by CBKSoft; It aims to provide comprehensive solutions for the entire information management lifecycle, from the creation, capture, indexing, storage, and retrieval to disposal of records and information assets, and for the automation and optimization of all paperwork or manual processes. It is used today with more than 300,000 users in different industries and sectors.[11]

M - Files

M-Files is an intelligent information management platform that helps companies make smart connections across the business and automate critical business processes, while at the same time keeping information secured and controlled. [12]

VIENNA Advantage

Enterprise level open source ERP solution with inbuilt CRM, document management system and business intelligence platform. Available on premises and on the cloud. Our customers are medium to large enterprises, multinational corporations and governments.

VIENNA Advantage is one unified business management suite supporting your transformation towards an intelligent enterprise. Leverage role-based access to critical data and streamline your processes across finance, procurement, manufacturing, HR, service, sales, marketing and more.[13]

Software Requirements Specification

Abstract

In Çankaya University, course transfers(or credit transfers) are made for three reasons in accordance with the regulations and directives; Undergraduate transfer, vertical transfer (via national DGS exam), and courses taken and completed at other universities (i.e., during summer school). During this period lots of request forms are signed and send to departments and all forms must be inspected by related instructor. Since all these operations are carried out by single person with in a limited time period, it is error prone . To minimize the mistakes due to nature of the process and reduce the overall processing time, there is a need for a web based course transfer automation system for the faculty which will help transfer coordinators manage course transfer process.

1. Introduction

1.1. Purpose of Document

This document is a software requirement feature for the web based Student Transfer Management System for Engineering Faculty (STMS) project. The document will not only define the product functions, user characteristics, constraints, and specific requirements of the system but also serve as a basis for the Software Design Document that will be prepared according to IEEE Std 1016-1998 [1]. The main objective of the project is a course transfer automation system that is needed for the faculty to help the transfer coordinators manage the course transfer process in order to minimize the errors arising from the nature of the process and reduce the total processing time.

1.2. Scope of Document

The purpose of this project is manage the student transfer system by using a web-based course transfer automation system for the faculty which will help transfer coordinators manage the course transfer process. According to this purpose, this system will be used by students who want to transfer, department coordinators to decide which courses will be checked by instructors and prepare forms for sending to instructors which contain courses that have a similar curriculum with Çankaya University courses and these forms are sent by coordinators to instructors. After these steps, instructors decide on the course equivalency. To perform the decision the system will send a link to the chosen instructors via email. The instructors can reach the system by clicking on the link. End of all processes, the department head checks the forms prepared by the coordinator and accepts or rejects the transfer of students. In

conclusion, the scope of our project is students, coordinators, instructors and heads of departments.

1.3. Definitions, Abbreviations, Acronyms

STMS: Student Transfer Management System for Engineering Faculty

IEEE: the Institute of Electrical and Electronics Engineers

SRS: Software Requirements Specification

DGS: Dikey Geçiş Sınavı

i.e. : id est, which is Latin for “that is.”

.NET: a framework that provides a programming guidelines that can be used to develop a wide range of applications

SQL: Structured Query Language

TCP : Transmission Control Protocol

IP: Internet Protocol

2. Overall Description

This section of the SRS describes the general factors that affect the Student Transfer Management System for Engineering Faculty and its requirements. This part provides a background for those requirements, which are defined in detail in Section 3 of the SRS, and makes them easier to understand.

2.1. Product Description

The STMS allows transfer coordinators to manage the course transfer process which leads to minimizing the mistakes and reducing the overall processing time. In Çankaya University, course transfers(or credit transfers) are made for three reasons in accordance with the regulations and directives; Undergraduate transfer, vertical transfer (via national DGS exam), and courses taken and completed at other universities (i.e., during summer school).With the Student Transfer Management System for Engineering Faculty during this period all the requested forms will no longer be submitted, processed and sent to departments physically.

2.1.1. System Interfaces

The system will access the email and personal information of the lecturers in the university from student affairs in order to integrate them into the database of the system.At the same time, the system will access the course curriculum information of the departments from student affairs in order to integrate it into the database.

2.1.2. User Interfaces

The system has 5 interfaces, such as admin, student, coordinator, lecturers and head of department.

The admin user is in charge of the system and gives the needed lecturers authorization within the system. The admin is not responsible with the student transfers.

The student creates an account and completes their profile by uploading a scanned version of their transcript, filling the course equivalence opinion request (coer) form, adding personal information such as phone number, e-mail etc. and information about their nationality to the STMS.

The coordinator is also required to create an account. After all the students have completed their course equivalence opinion request (coer) form and their transcript information the coordinator is able to examine the requests by screening all the individual students information. On one side of the screen the coordinator is able to view the students transcript and request form and on the other side of the screen he/she can create the list of courses according to the transcript, course name, course code and course credit. The finished course list will be sended to the head of department to review and submit the lists to the respective departments. The coordinators are able to review all the action statuses within the system and send reminders if needed.

The lecturers are not required to create an account since they won't use the system as much as the coordinator. For easy access to the STMS after the department heads send the lists of courses to the departments, lecturers will receive an email with a time-limited link that provides access to the system. The lecturers are required to review the courses for course equivalency and approve or deny the request with the help of buttons.

After all lecturers review the course equivalencies, the coordinator reviews the final course list and the head of department reviews and approves the final request forms.The head of department can review all the

actions within the system and inspect the taken actions. As an output the system creates a list of the approved courses.

2.1.3. Hardware Interfaces

For users, there are no hardware interfaces required to run the software other than a computer capable of serving and displaying web pages.

2.1.4. Software Interfaces

The system will be hosted on .NET Angular JS WINDOWS Hosting. The system depends on SQL for persistent data storage. For serving web pages and communicating with databases, .NETFramework version 3.5 is used.

2.1.5. Communication Interfaces

An internet connection is required to run the system. The default communication protocol, encrypted communication control protocol, TCP / IP Internet protocol, the basic protocol provided by Windows Server and used between the web server with port 21 and the client will be used.

2.1.6. Operations

- The admin user shall be able to authorize required lecturers with user roles as required.
- The student users shall be able to fill the course equivalence opinion request (coer) form, upload their transcript, enter personal data.
- The coordinator users shall be able to assign an assistant for themselves in the system, create a list of courses for each student and view personal information of students.
- The lecturer users shall be able to review the course requests and approve or deny them.
- The head of department shall be able to review all students' information and review the actions of the coordinator and lecturers.

2.2. User Characteristic

Student:

- Students shall be able to register to the system.
- Students shall be able to upload their transcripts to the system.

Coordinator:

- The coordinator shall be able to register to the system.
- The coordinator shall be able to see the students' appeal and transcript.
- The coordinator shall be able to send these documents to the related departments head according to the departments of courses in the transcript.
- The coordinator shall be able to take the assessments' results from instructors thanks to this the coordinator shall be able to keep up with the process.
- The coordinator shall be able to create a table that contains accepted courses for the transfer and send it to the departments head.

Department head:

- The department head shall be able to take the documents from the coordinator and approve them.
- The department head shall be able to decide which instructors will check the courses.
- The department head shall be able to see the accepted courses list and accept or reject the list.

Instructors:

- The instructors shall be able to reach the system if they get a link with an email.
- The instructors shall be able to accept or reject the related courses.

Admin:

- The admin shall be able to modify the system.
- The admin shall be able to identify to roles of users.

2.3. Constraints

The software development team obeys the IEEE standards [1, 2, 3, 4, 5, 6] for the software development process stated in the references section.

The system is accessible for only department coordinators, instructors and department heads. Department coordinators and department heads must have a registered e-mail and password. The instructors can only access the system with a link sent by the coordinator. The system only works on a web browser.

2.4. Assumptions and Dependencies

It is assumed that each user logs into the system via a device and a browser that can actively connect to the internet and run various script codes.

2. Specific Requirements

3.1. External Interfaces

The system shall be able to get the departments course curriculum information and the email and personal information of the lecturers in the university from student affairs in order to integrate them into the database of the system.

3.2. Functional Requirements

Use Case Name: Register

Actors: Student, User(Dean, Coordinator, Head of department)

Goal: Creating an account for users

Pre-Condition:

- Validity of Student Certificate

Post-Condition:

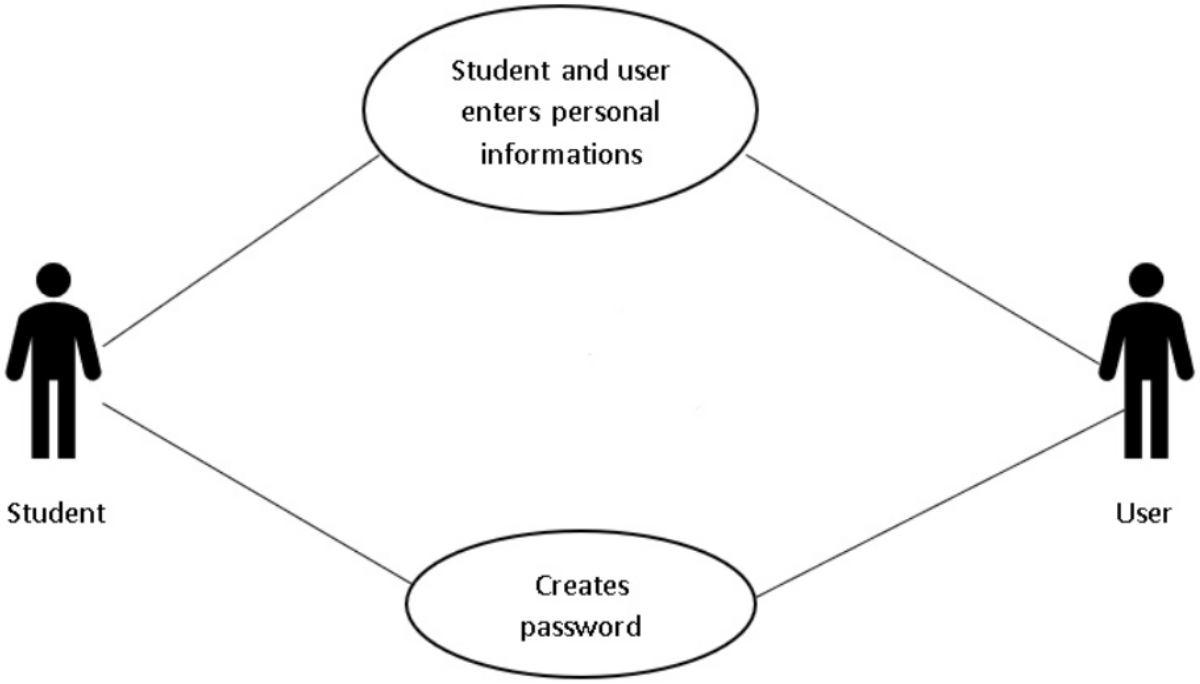
- System saves student’s and user’s information.
- Student and user have an account on the system.

Exceptions:

- Student and user can enter an invalid email address or department information.
- Account can be already exists.

Main Flow:

- Student and user enter their name, surname, e-mail address information.
- If the registered person is a student, he/she uploads his/her school information, which department he/she studied.
- Student and user create a password.



Use Case Name: Login

Actors: User (Student, Dean, Coordinator, Head of department)

Goal: Users login to the system with their account information

Pre-Condition:

- Whether the user is registered in the system

Post-Condition:

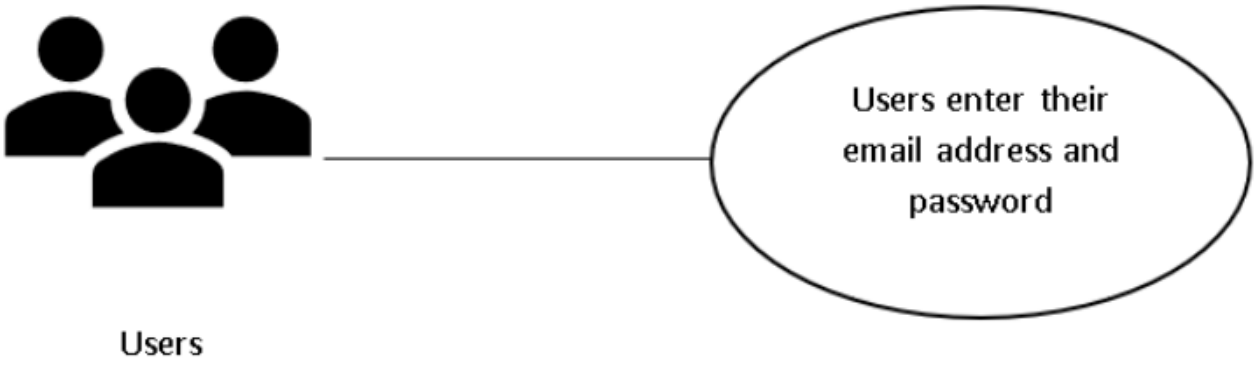
- The user logs into the system.
- The user gains access to the system.

Exceptions:

- The user information entered may be incomplete or incorrect.

Main Flow:

- User enters e-mail and password information



Use Case Name: Sort out accepted courses

Actors: Coordinator

Goal: Sort out the accepted courses and sending the accepted courses to the relevant department heads

Pre-Condition:

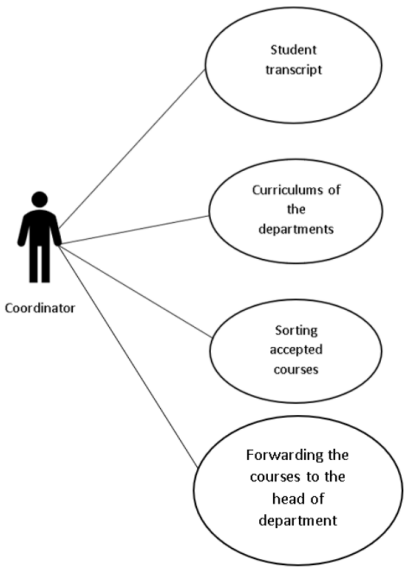
- Coordinator accesses students' transcripts
- Coordinator reaches the curricula of the departments in the school

Post-Condition:

- Department heads access accepted courses information

Main Flow:

- The coordinator accesses the transcripts of the students thanks to the system
- The coordinator reaches the curricula of the relevant departments.
- The transcripts and the courses in the curriculum are compared and sorted by the coordinator.
- Accepted courses are delivered to the relevant department heads.



Use Case Name: Listing approved courses

Actors: Coordinator,

Goal: Coordinators list the courses approved by the instructors

Pre-Condition:

- Lessons should be assessed by instructors

Post-Condition:

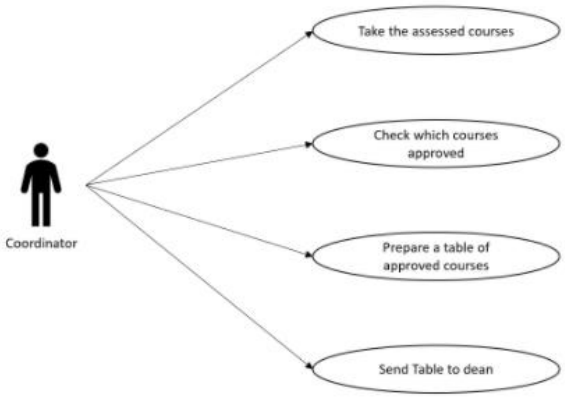
- Acceptance or rejection of the request

Exceptions:

- Instructors forget to control the courses

Main Flow:

- The coordinator takes the assessed courses
- The coordinator tabulates the courses which approved by instructors
- The coordinator send the table to the dean.



Use Case Name: Countdown

Actors: System

Goal:To check if the emails sent to the instructors are answered within a certain period of time.

Pre-Condition:

- Department heads should choose instructors
- System should send e-mail to instructors

Post-Condition:

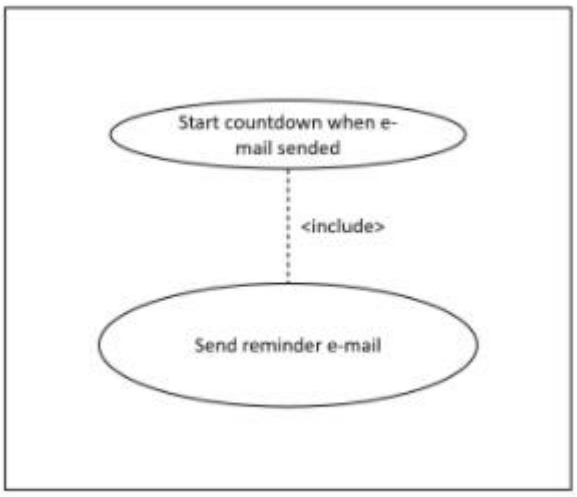
- If there is no response within the specified day, the system will send a reminder e-mail.

Exceptions:

- If the instructors do not receive an email
- If email is sent to the wrong instructor

Main Flow:

- The system sends an e-mail to the selected instructors about the courses it will check.
- The countdown will start after the mail is sent.
- If there is no response during the countdown, the system sends a reminder e-mail.
- Countdown starts again



Use Case Name: Mail

Actors: Instructors, Department Heads

Goal: Department heads choose the instructors which is going to check the lessons then system send them a link via e-mail

Pre-Condition:

- Choosing the instructors to control the courses by the head of the department
- Department heads must reach the instructors database

Post-Condition:

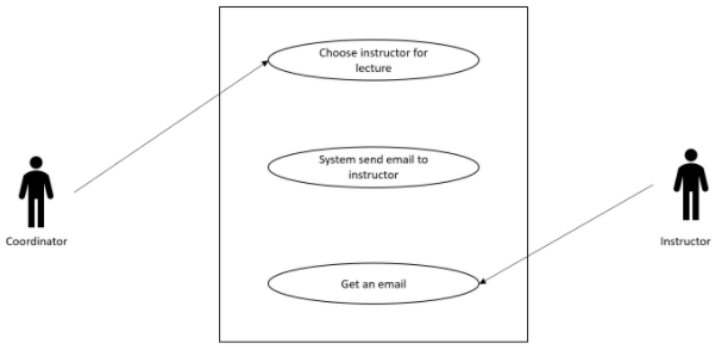
- Lessons controlled by the instructors

Exceptions:

- Department heads can choose the wrong instructors for courses

Main Flow:

- The department heads take the tables of courses which should be approved by instructors
- Head of the department selects suitable instructors to check the courses on the list
- The system sends an e-mail to the selected instructors about the courses it will check



Use Case Name: Student Document Upload

Actors: Student

Goal: Uploading scanned version of their Transcript and filling the course equivalence opinion request (coer) form.

Pre- Condition:

- Students shall be logged in to the system

Post- Condition:

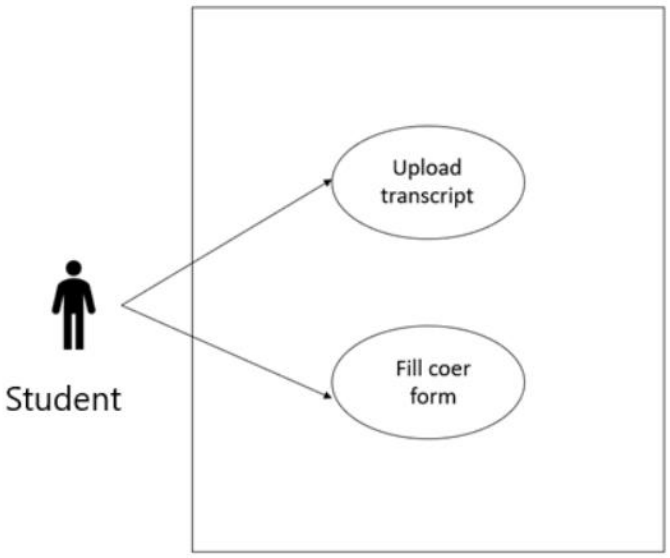
- Coordinator, head of department and lecturers shall be able to view the documents

Exceptions:

- The scanned transcript shall at least have the resolution of 300dpi

Main Flow:

- Student shall upload their pre scanned transcript to the upload field
- Student shall fill the course equivalence opinion request (coer) form



Use Case Name: Defining User Roles

Actors: Admin

Goal: The admin shall be able to authorize lecturers within the university for user roles in the system.

Pre-Condition:

- *The admin shall be pre-determined

Post- Condition:

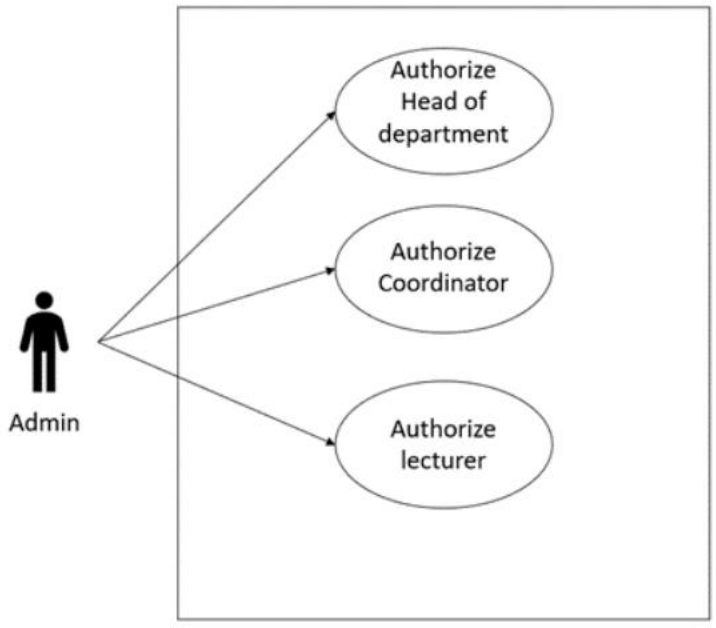
- Coordinator, head of department and lecturers' roles shall all be determined.

Exceptions:

- Empty roles shall not be allowed

Main Flow:

- The admin shall fist authorize the head of department roles
- The coordinator roles shall be determined
- The lecturer roles shall be authorized



Use Case Name: Reviewing Course Requests

Actors: Lecturers

Goal: The lecturer shall be able to accept or reject each course equivalence request for each student.

Pre- Condition:

- The lecturer shall receive a email with a link to provide access to the system.

Post- Condition:

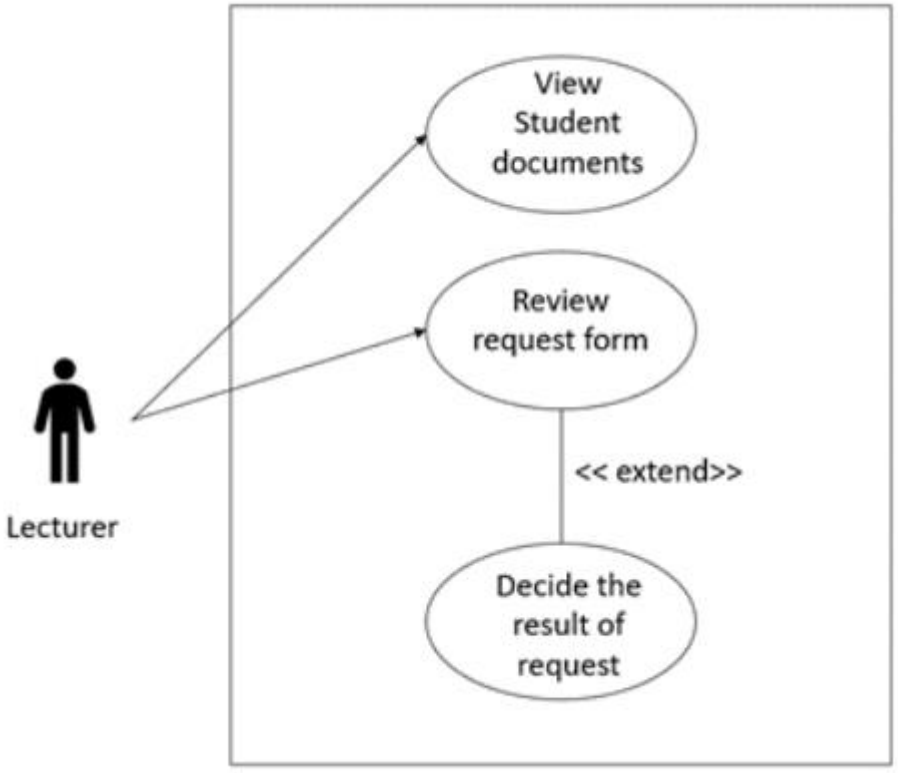
- The head of department shall accept the revisions of the lecturers.

Exceptions:

- The revisions shall be open for change.

Main Flow:

- The lecturer shall review the course equivalence request of the according student
- The lecturer shall be able to view the documents of the students .
- The lecturer shall decide whether to accept or reject the request.



User Case Name: Courses Database

Actors: Admin,Coordinators,Instructors

Goal: Keeping the necessary information about the courses in the system so that the coordinators and instructors can access the resources to check the equivalence status of the courses.

Pre-Condition:

- Admin must reach the school database for taking courses syllabus.

Post-Condition:

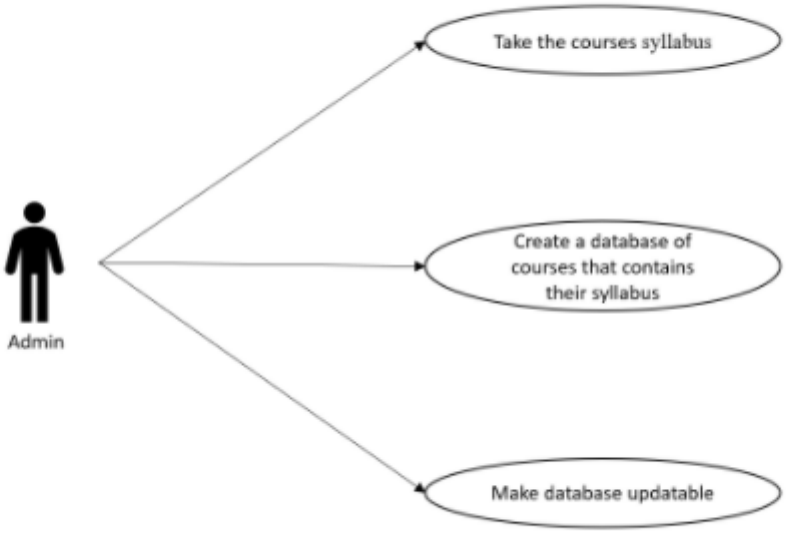
- Coordinator and instructors must reach the courses database.

Exceptions:

- The content of the courses may change.

Main Flow:

- Admin take the courses syllabus from school database.
- Make a courses database which reachable from instructors and coordinators.
- If courses content change admin can update the courses



User Case Name: Instructor Database

Actors: Admin,Department Heads

Goal: Keeping the necessary information about the instructors in the system so that the departments head can choose appropriate instructors to check courses

Pre-Condition:

- Admin must reach the school database for taking instructors' information.

Post-Condition:

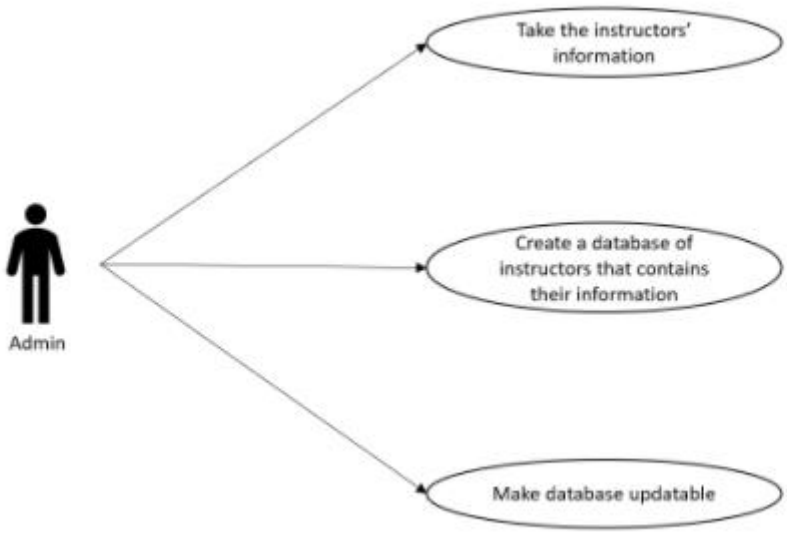
- Department Heads must reach the courses database.

Exceptions:

- Instructors may leave school or new instructors may come to school

Main Flow:

- Admin take the instructors' information from school database.
- Make a instructor database which reachable from department heads
- If new instructors will come or some instructors leave, the admin can update the database.



User Case Name: Students Database

Actors: System,students

Goal: Keeping the necessary information about the students in the system

Pre-Condition:

- Students must register to the system

Post-Condition:

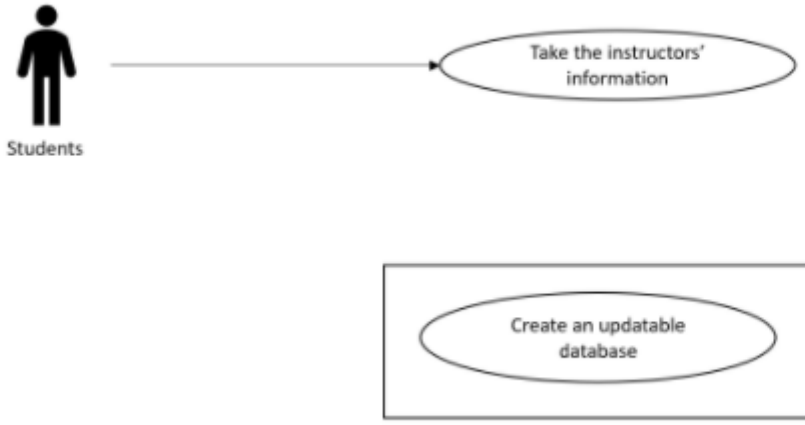
- Department Heads, deans, coordinators must reach the student database.

Exceptions:

- Students may enter their information incorrectly

Main Flow:

- Students must create an account by filling out the questions given during registration.
- The system should store the entered information by creating an updatable database.



3.3. Database Requirements

In this project, we will obtain most of the required data from the student affairs office. First, the student data which are the student's personal information, contact information and transcript of student. As students, we need instructor data because coordinators should have the instructors' contact and department information for assigning the control process. To decide course equivalency, instructors will need transcripts of courses of the Çankaya University. Because of this reason, we have to create data of transcript, we will obtain this data from students' affairs office too.

3.4. Design Constraints

The Intuitive design of the system will be developed according to the usability topics; visibility of system status, user control and freedom, consistency and standards, efficiency of use, aesthetic and minimalist design, help and documentation and error handling are the focus points to develop a user friendly system.

3.5. Software System Attributes

3.5.1. Reliability

The system is intended to be one hundred percent reliable. The system can be updated at any time. This way, reliability is ensured.

3.5.2. Availability

The user will be able to access the system whenever the device has an internet connection.

3.5.3. Security

Not every system user is allowed to access the information in the database. Only authorized users can access documents. The information of the users will not be shared with the third party application.

3.5.4. Maintainability

The system can be updated taking into account the notifications from the users, but the operation of the system can never be changed. Documents in the system will be removed from the system database at the end of each semester.

3.6.5. Portability

The system will be supported by operating systems on all devices. The project does not require any external hardware. All database records will be processed on this server.In this way, our system will be portable.

Software Design Document

1. Introduction

1.1. Purpose of Document

The purpose of this project is manage the student transfer system by using a web-based course transfer

automation system for the faculty which will help transfer coordinators manage the course transfer process. Our system aims to prevent the problems of losing documents and forgetting the tasks that need to be done and conduct the student transfer management easier, faster, and safer.

1.2. Scope of Document

According to the purpose, this system will be used by students who want to transfer, department coordinators to decide which courses will be checked by instructors and prepare forms for sending to instructors which contain courses that have a similar curriculum with Çankaya University courses and these forms are sent by coordinators to instructors. After these steps, instructors decide on the course equivalency. To perform the decision the system will send a link to the chosen instructors via email. The instructors can reach the system by clicking on the link. End of all processes, the department head checks the forms prepared by the coordinator and accepts or rejects the transfer of students. In conclusion, the scope of our project is students, coordinators, instructors and heads of departments.

2. Design Considerations

2.1 Approach

- The transcripts and forms uploaded by the students in the registration section will appear in front of the coordinator.
- The coordinator compares the course syllabus registered in the system with the forms uploaded by the students, and saves the list on the screen for the approval of the dean.
- The dean completes the approval process by examining the information of the student registered in the system by the coordinator.
- After the approval of the dean, the system sends an e-mail to the relevant department lecturers.
- Along with e-mail, lecturers provide access to the system.
- The courses approved by the lecturers are on the dean's page.
- The dean checks and approves the action from the lecturers.
- After the approval of the dean, the curriculum is created for the student.

2.2 Tools Used

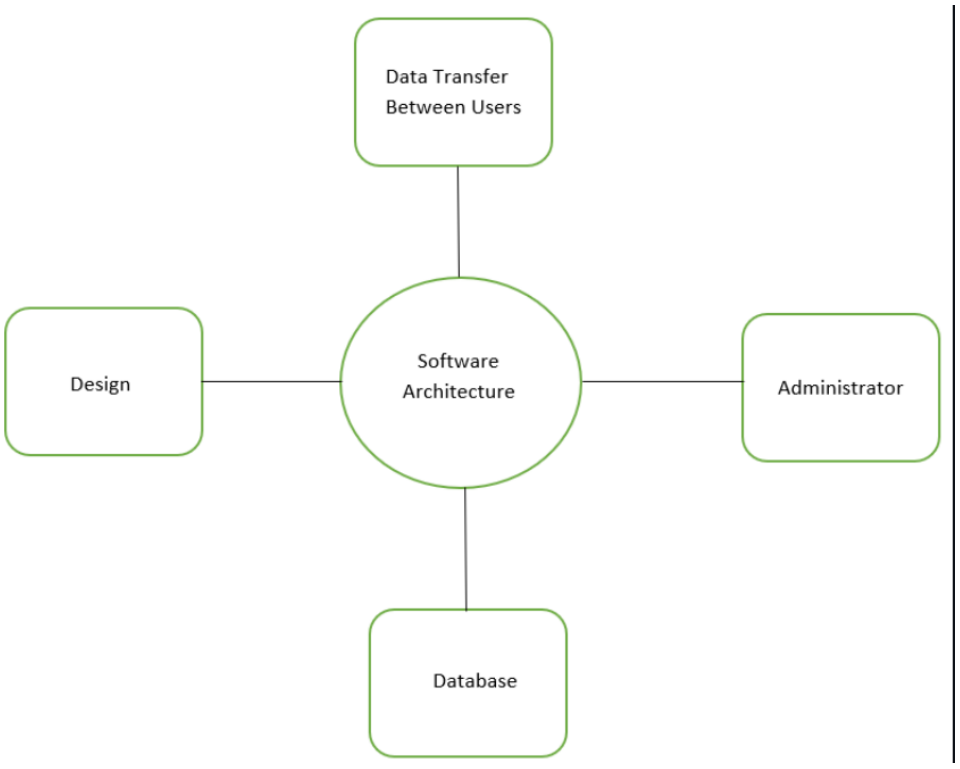
- The system has many different interfaces for student transfer management system users. It has all the necessary elements to control the application process.

- Microsoft Visual Studio will be used to lay the foundation of our web page.Microsoft Visual Studio is an IDE for Windows with an integrated development environment.
- Ms – SQL will be used to store and update information on our website.Ms-SQL is a database type produced by Microsoft.
- AngularJS will be used so that the information in the web page can be easily integrated into the site. Angular JS is a structural framework for dynamic web applications. It allows the use of HTML as the template language.
- Html, Css and JavaScript will be used for front-end design of the web page.HTML lets you add content to a web page. CSS determines the design, style and layout of web pages. JavaScript manages the behavior of web pages.
-

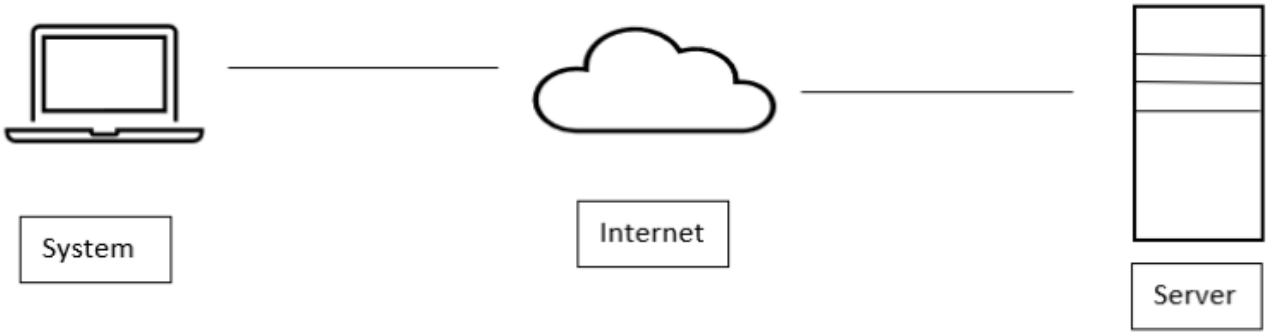
3. Architecture

4.

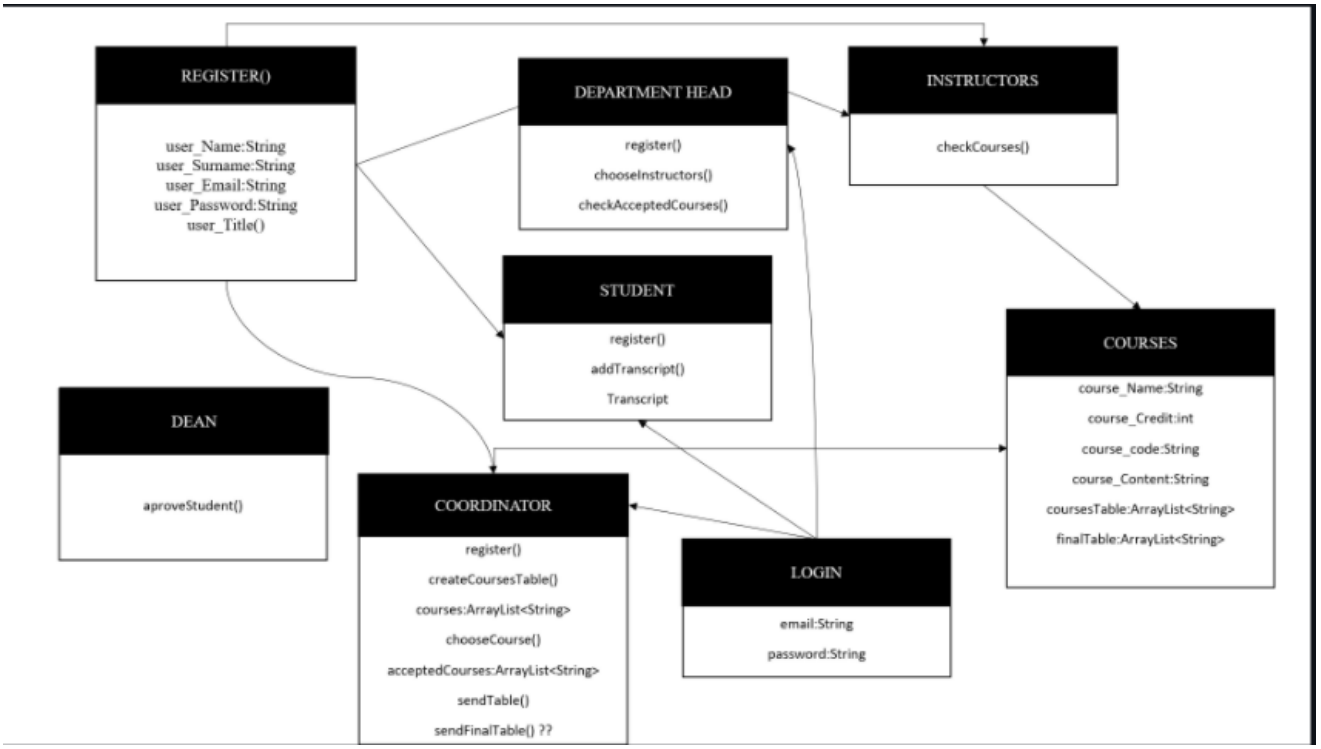
3.1 Software Architecture



3.2 Hardware Architecture



3.3 ER Diagram



4. System Interface

The system utilizes MSSQL 15 for long-term data storage. To provide web pages and communicate with databases, Entity Framework version 6.1 is used. For.NET applications, Entity Framework is an object-relational mapping framework. Object-relational mapping allows object-oriented programming languages to use database queries and operations. The system will come with a SQL definition file that details the initial configuration of the data tables, which will be utilized by the agency's administrator to set up persistent data storage.

4.1 External System Interface

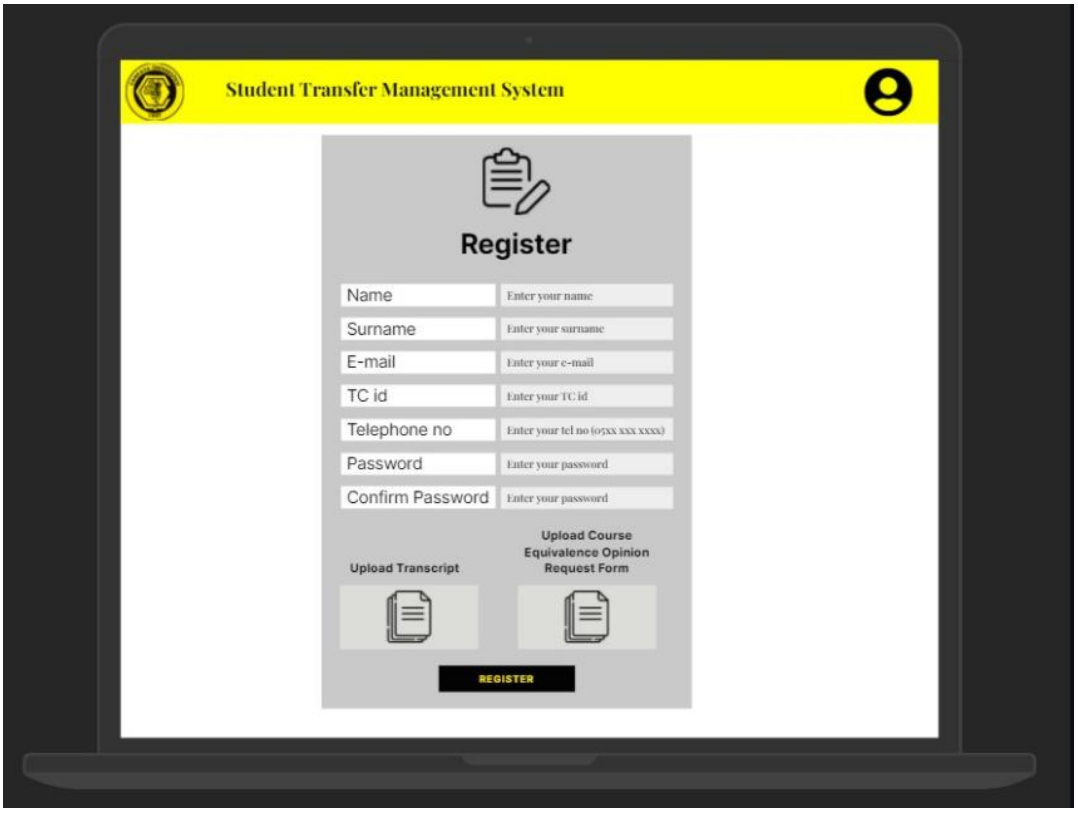
The system shall be able to get the departments course curriculum information and the email and personal information of the lecturers in the university from student affairs in order to integrate them into the database of the system.

5. Screen Definitions

5.1 Register Interfaces

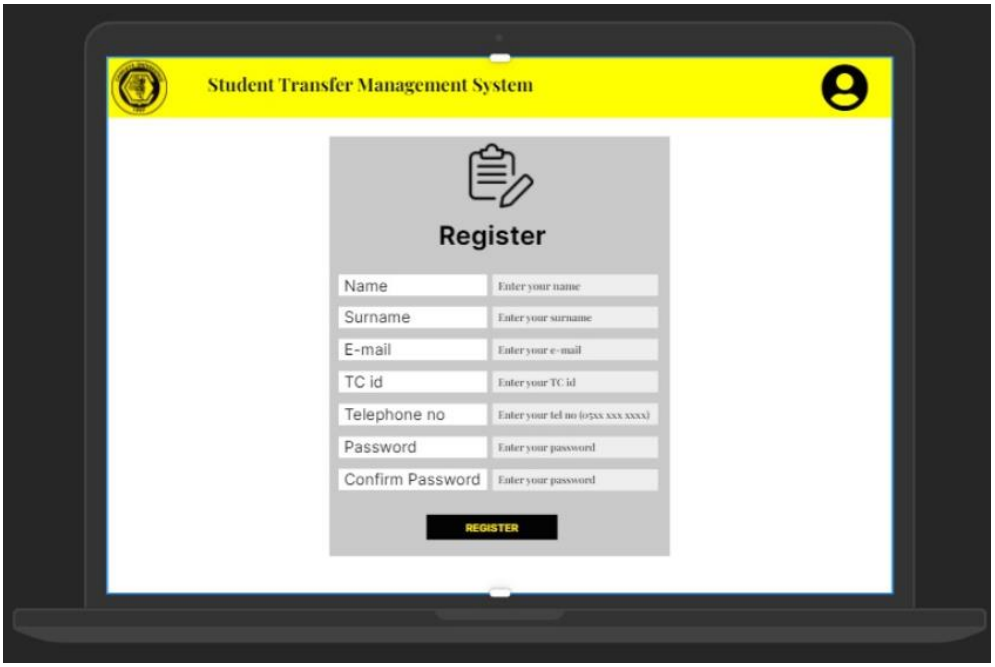
Student Register

Students shall be able to register to the system. There are four input areas which for name, surname, e-mail and creating password and a document upload area for students' transcript for that purpose and a button to register.



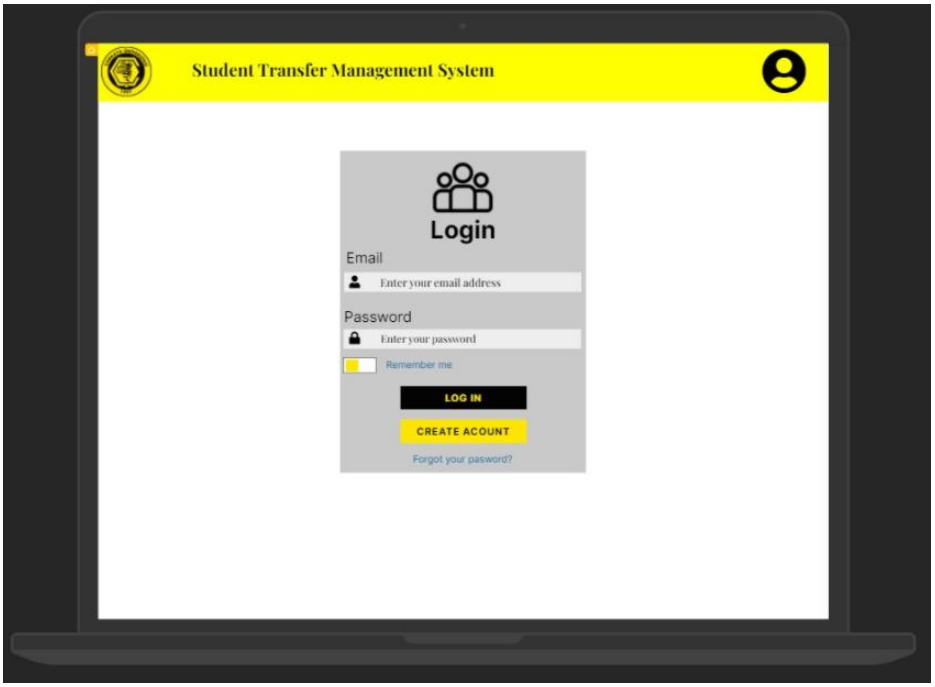
Register for Coordinator and Department Head

Coordinator and department head shall be able to register to the system. There are four input areas which for name, surname, e-mail and creating password. Also there is a checkbox for choose a title which are coordinator or department head and a button to register.



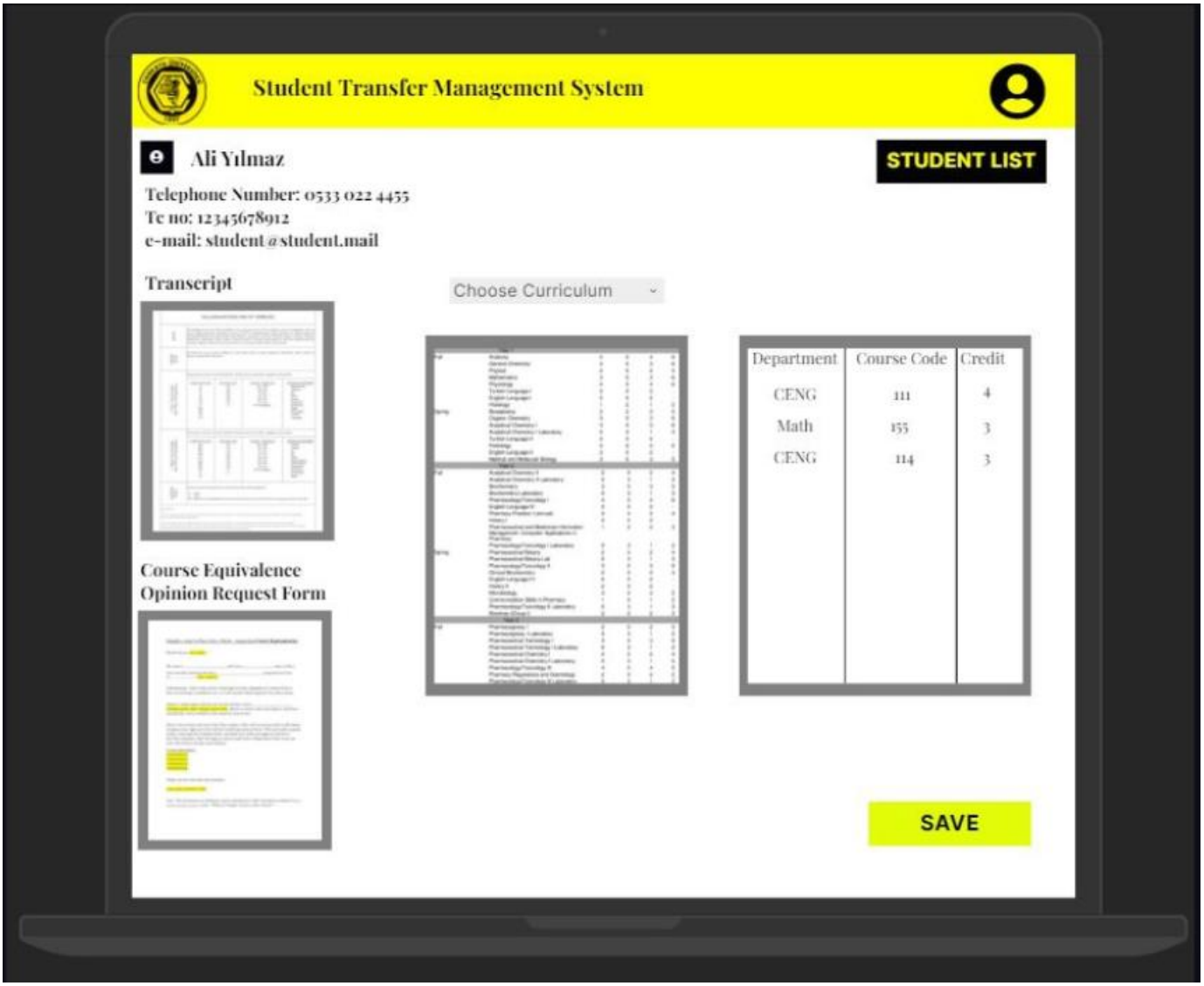
5.2 Login

Accounts contain a username and a password. There are two input areas for that purpose and a button to log in. Accounts can be created by students, coordinators and the department heads. The logo is shown in the login page top of the input boxes.



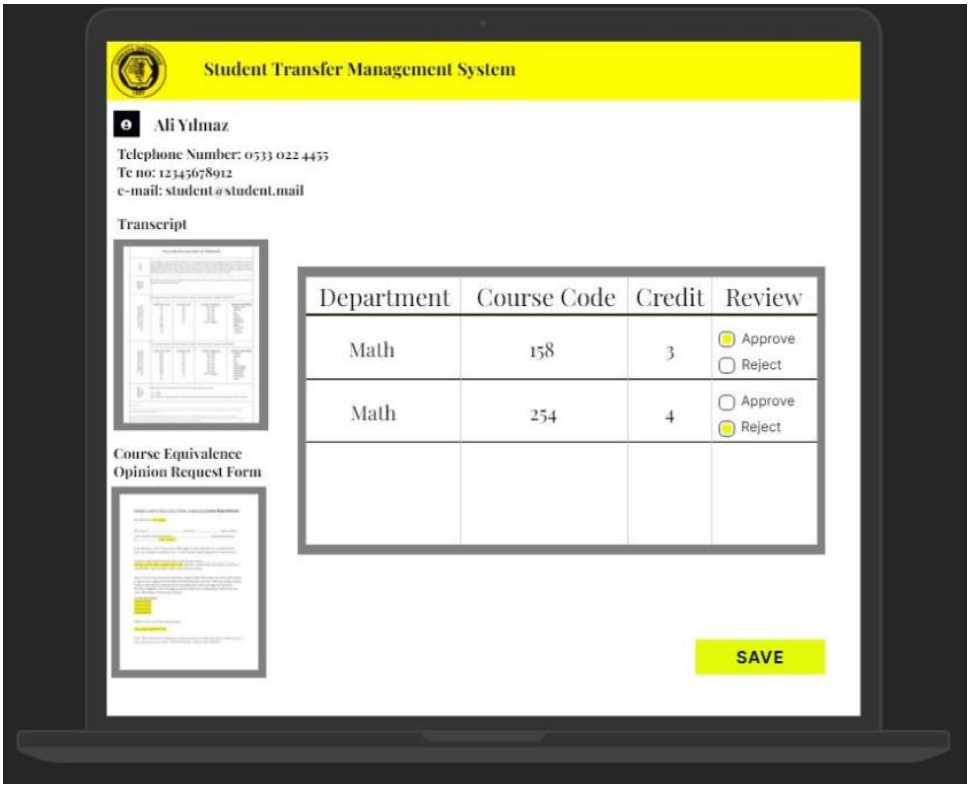
5.3 Coordinator Interfaces

On the coordinator screen, there is information about the applicant, the transcript of the applicant, and the school curriculum. There is also a panel on the coordinator screen where the departments and departments can select the department heads and send them the relevant courses.



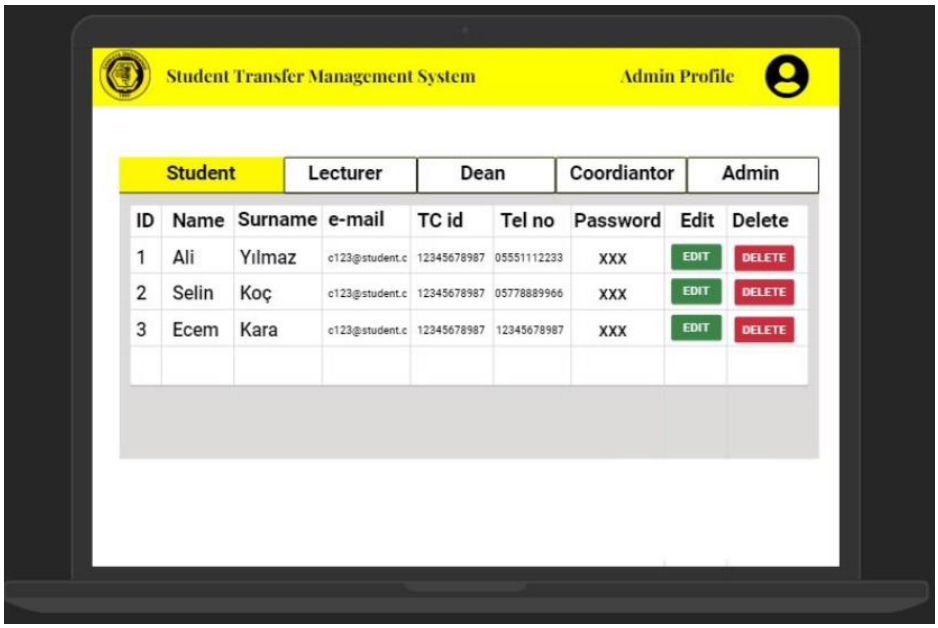
5.4 Lecturer Interface

There are 2 courses on the screen of the lecturer, one of which is the course that the applicant student has taken at his/her university and the other is the course that is thought to be similar in our university. As a result of this evaluation, there is also an area where it marks whether it is accepted or not.



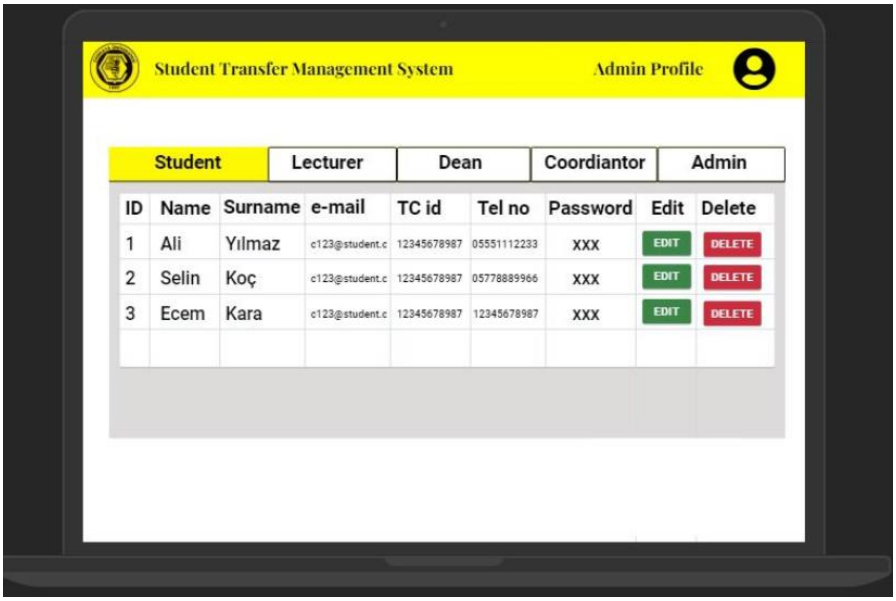
5.5 Admin Interface

There are necessary areas where it can perform operations such as making necessary interventions to the system and determining roles.



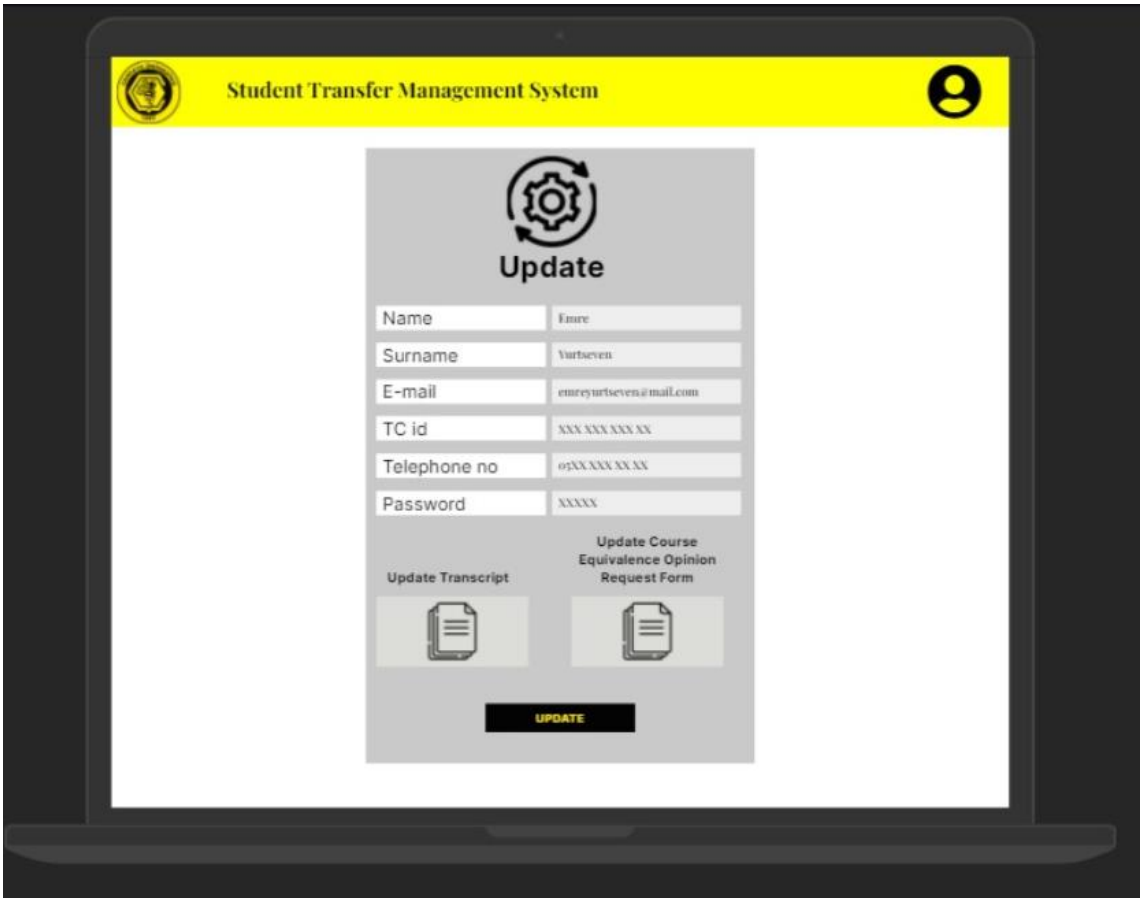
5.6 Dean Interface

The Dean's screen includes the most recently accepted courses and student information. There is also a selection area for the dean to admit the student to the school.



5.7 Update Interface

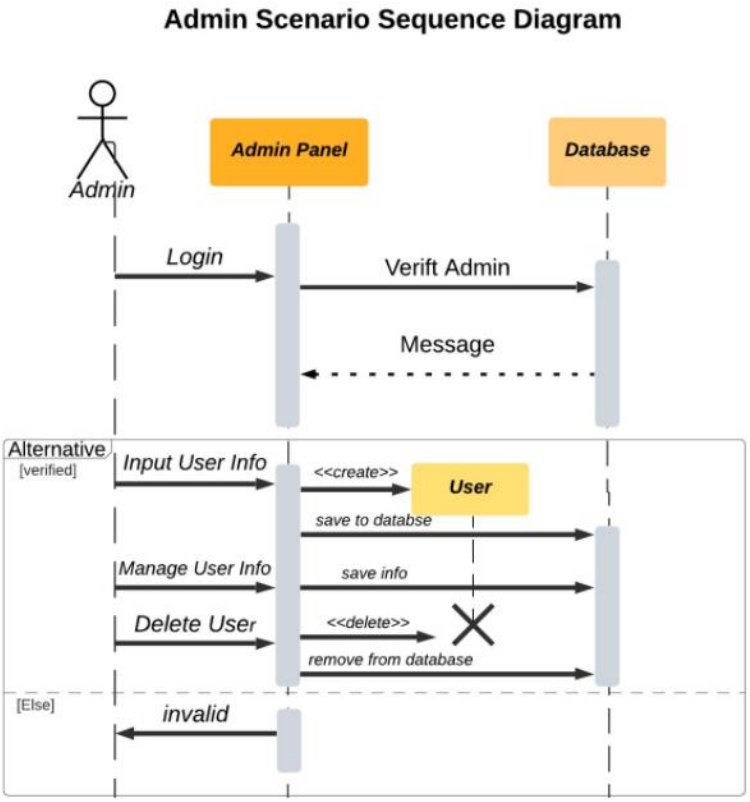
If there is any change in the information entered during registration, there are places to fill in the required fields again and save them.



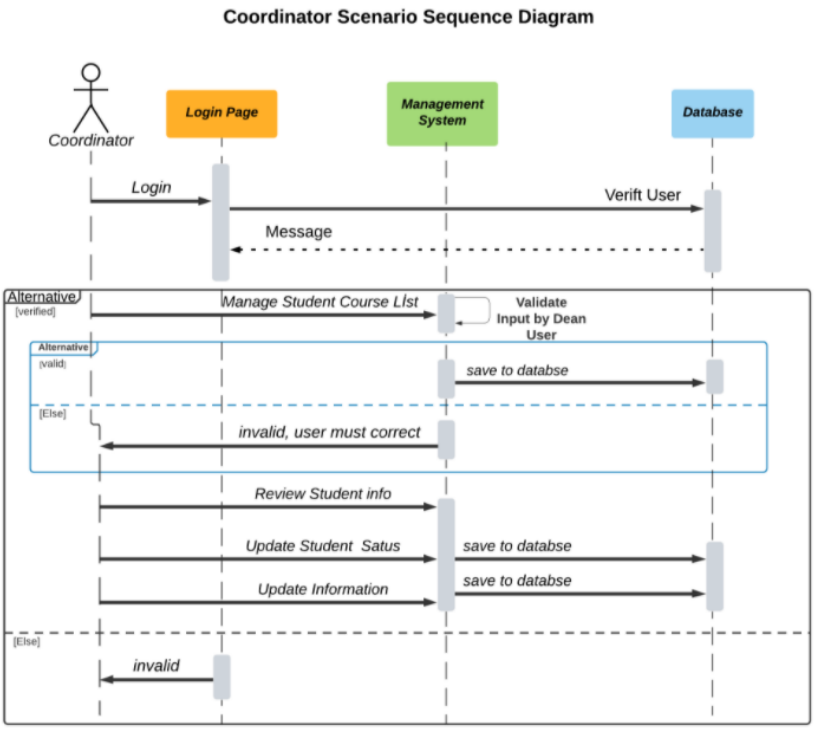
6. Process Design

6.1 Sequence Diagrams

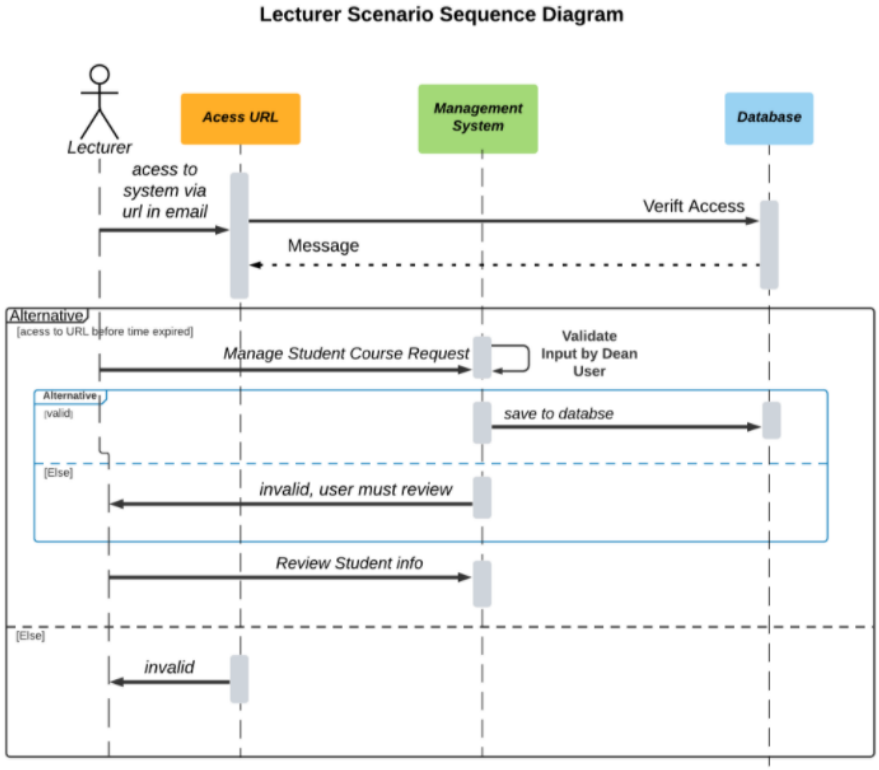
6.1.1 Admin Scenario Sequence Diagram



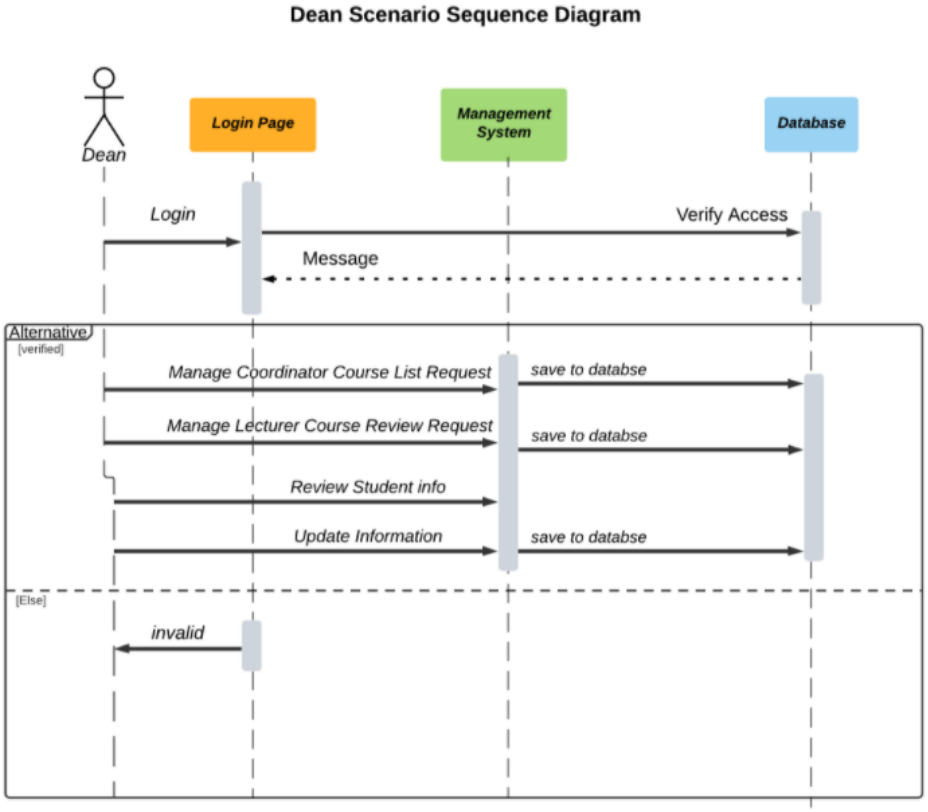
6.1.2 Coordinator Scenario Sequence Diagram



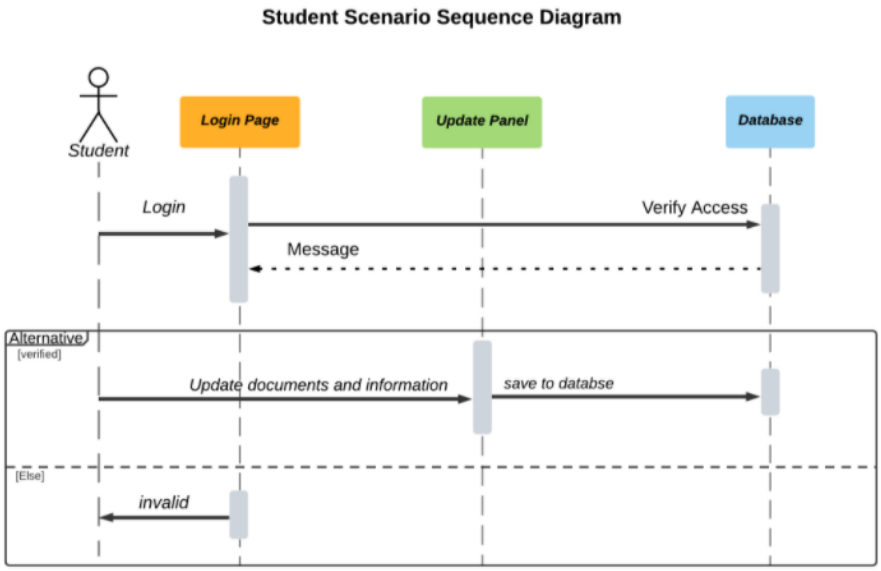
6.1.3 Lecturer Scenario Sequence Diagram



6.1.4 Dean Scenario Sequence Diagram



6.1.5 Student Scenario Sequence Diagram



7. Database Design

7.1 Student Information

| Attribute Name| Type | Description |

| ---------- | ---------- |---------- |

| Student_ID | AutoField |Auto generated positive integer id for table entry |

| email | nvarchar |Email address for student user |

| first_name | nvarchar |First Name of Student User|

| last_name | nvarchar |Last Name of Student User |

| password | nvarchar |Password of Student User|

| TC_no | int | Turkish Identification Number of Student User|

| Tel_no | int | Telephone number of Student User|

| transcript | varchar(max) | The scanned transcript in pdf format of student user|

| document | varchar(max) | The scanned Course Equivalance Opinion Request Form in pdf format of student user|

7.2 Coordinator Information

| Attribute Name | Type | Description |

| ---------- | ---------- |---------- |

| Coordinator_ID | AutoField |Auto generated positive integer id for table entry |

| email | nvarchar |Email address for coordinator user |

| first_name | nvarchar |First Name of Coordinator User|

| last_name | nvarchar |Last Name of Coordinato User |

| password | nvarchar |Password of Coordinator User|

| TC_no | int | Turkish Identification Number of Coordinator User|

| Tel_no | int | Telephone number of Coordinator User|

7.3 Dean Information

| Attribute Name | Type | Description |

| ---------- | ---------- |---------- |

| Dean_ID | AutoField |Auto generated positive integer id for table entry |

| email | nvarchar |Email address for dean user |

| first_name | nvarchar |First Name of dean User|

| last_name | nvarchar |Last Name of dean User |

| password | nvarchar |Password of dean User|

| TC_no | int | Turkish Identification Number of dean User|

| Tel_no | int | Telephone number of dean User|

7.4 Admin

| Attribute Name | Type | Description |

| ---------- | ---------- |---------- |

| Admin_ID | AutoField |Auto generated positive integer id for table entry |

| email | nvarchar |Email address for dean user |

| first_name | nvarchar |First Name of admin User|

| last_name | nvarchar |Last Name of admin User |

| password | nvarchar |Password of admin User|

| Dean_ID | ForeignKey(id) |Foreign Key to Dean Information table |

| Coordinator_ID | ForeignKey(id) |Foreign Key to Coordinator Information table |

| Student_ID | ForeignKey(id) |Foreign Key to Student Information table |

| Lecturer_ID | ForeignKey(id) |Foreign Key to Lecturer table |

7.5 Coordinator

| Attribute Name | Type | Description |

| ---------- | ---------- |---------- |

| Coordinator_ID | ForeignKey(id) |Foreign Key to Coordinator Information table |

| Student_ID | ForeignKey(id) |Foreign Key to Student Information table |

| Course_ID | ForeignKey(id) |Foreign Key to Course table |

| Liste_ID | ForeignKey(id) |Foreign Key to Liste table |

| Student_Status | Boolean |Status of the Course List Status|

7.6 Dean

| Attribute Name | Type | Description |

| ---------- | ---------- |---------- |

| Dean_ID | ForeignKey(id) |Foreign Key to Dean Information table |

| Student_ID | ForeignKey(id) |Foreign Key to Student Information table |

| Liste_ID | ForeignKey(id) |Foreign Key to Liste table |

| Approve_List | Boolean |Approve or Rejection status of the Course List |

7.7 Lecturer

| Attribute Name | Type | Description |

| ---------- | ---------- |---------- |

| Lecturer_ID | AutoField |Auto generated positive integer id for table entry |

| email | nvarchar |Email address for dean user |

| first_name | nvarchar |First Name of lecturer User|

| last_name | nvarchar |Last Name of lecturer User |

| department | nvarchar |Department of lecturer User |

| Student_ID | ForeignKey(id) |Foreign Key to Student Information table |

| Course_ID | ForeignKey(id) |Foreign Key to Course table |

| Liste_ID | ForeignKey(id) |Foreign Key to Liste table |

| Approve_Request| Boolean |Approve or Rejection status of the Requested Course |

7.8 Course

| Attribute Name | Type | Description |

| ---------- | ---------- |---------- |

| Course_ID | AutoField |Auto generated positive integer id for table entry |

| department | nvarchar |deaprtment information for course schedule |

| Course_Code | nvarchar |Course code information|

| Course_Name | nvarchar |Course name information|

| Course_Type | nvarchar |Course type information |

| Course_Duration| nvarchar |Course duration information |

| Local_Credit | int |Course local credit information|

| AKTS | int |Course European Credit Accumulation and Transfer System information|

7.9 liste

| Attribute Name | Type | Description |

| ---------- | ---------- |---------- |

| Liste_ID | AutoField |Auto generated positive integer id for table entry |

| department | nvarchar |deaprtment information for course schedule |

| Course_Code | nvarchar |Course code information|

| Local_Credit | int |Course local credit information|

Discussion

In this section, we will mention the possible problems of the system and possible difficulties in the development phase.

The Student Transfer Management System for Engineering Faculty will receive some information from the student affairs of our university, and in line with this data, the system will include the syllabus and the information of the faculty members. Student affairs and the systems database we will develop must be compatible with each other and properly integrated. The system needs many different user roles to work properly and we think the biggest challenge we will face will be to integrate them properly. Many users mean many different user role descriptions and functions, in addition, the pages and functions that each user type can access will be different from each other. Most of the processes in our system should be open to update for possible errors that the users of the system we will develop can make during the process. The system should be suitable for updating, not only for possible errors that users can make, but also for information that needs to be updated or changed over time.

Conclusion

In summary the fisrt stage of the project was the literature review, where data was collected to examining similar projects and documents made so far. Based on these researches the literature review was created. One of the important documents that was needed to prepare after the literature review was the software requirements specifications document. The software requirements specifications document outlines the functions and purpose of the future software product, what the system will do and how it will perform. Another important document is the software design document. The software design document is written to guide the software developer team through the architecture of the software project. It includes insights about system architecture and details about the user interface design, also information about the database is given. The Project Report includes all the theoretical information about the software system. After this stage, the plan is to carry out implementations of the Student Transfer management System for Engineering Faculty based on the theoretical documents that have been prepared.

References

[1] https://dergipark.org.tr/tr/download/article-file/454324

[2] https://www.webtekno.com/microsoft-visual-studio-nedir-h92228.html

[3] https://www.argenova.com.tr/angular-js-nedir-tercih-etmeli-miyiz

[4] https://ofisdata.com/sql-nedir

[5] https://teknodestek.com.tr/optical-character-recognition-ocr-nedir/

[6] https://www.betabreakers.com/what-is-an-intuitive-interface/

[7] https://tr.phhsnews.com/articles/uiux/intuitive-ui-how-it-can-help-improve-ux.html

[8] https://uxplanet.org/create-visual-ui-design-for-better-products-14f91e557638

[9]https://avantgardeoftware.com/?gclid=CjwKCAjwoP6LBhBlEiwAvCcthLO_3NuiwHuFjYSs_C5RpHtop46cYsHlr2cs-MQ72YZ34poLJMlCbBoC1eIQAvD_BwE

[10] https://ebysportal.saglik.gov.tr/TR,2006/elektronik-belge-yonetim-sistemi-nedir.html

[11] https://envision.infofort.com/tr/about/

[12] https://www.m-files.com

[13] https://www.viennaadvantage.com

[14] https://www.edrawsoft.com/article/use-case-diagram-examples.html.

[15] https://www.geeksforgeeks.org/designing-use-cases-for-a-project/.