

Software Requirements Specification

Automated Graduation Management System

Table of Contents

- 1. Introduction
- 2. Overall Description
- 3. Specific Requirements

Revision History

Version	Date	Description	Author
1.1	2024-11-24	Before the review	System Analyst

1. Introduction

1.1 Purpose

The purpose of this Software Requirements Specification (SRS) document is to define the functional and non-functional requirements for the development of the Graduation Management System. This system is intended to automate and optimize the manual processes currently used at IZTECH for managing the graduation requirements of undergraduate students.

The system will provide a centralized platform that streamlines interactions between students, academic advisors, and administrative staff. By replacing manual workflows with automated processes, the system aims to achieve the following objectives:

- 1. **Improve Efficiency:** Automate repetitive and time-consuming tasks such as credit validation, graduation application submission, and approval workflows.
- 2. **Enhance Accuracy:** Reduce errors arising from manual data handling by providing automated checks and validation mechanisms.
- 3. **Ensure Transparency:** Offer a clear and accessible platform for students, advisors, and staff to monitor the progress of graduation applications and related activities.

4. **Simplify Communication:** Facilitate seamless communication between stakeholders involved in the graduation process, ensuring that students receive timely feedback and updates on their applications.
5. **Adherence to Policies:** Ensure compliance with IZTECH's academic policies and procedures by integrating them into the system's logic and workflows.

This document will serve as a comprehensive guide for the development team, stakeholders, and testers, outlining all necessary requirements to ensure the system aligns with the intended goals and delivers value to its users.

1.2 Scope

The Automated Graduation Management System (AGMS) is a web-based application designed to replace manual graduation workflows at IZTECH. The system will:

- Automate graduation application submissions, transcript validation, and approval workflows
- Enable real-time tracking of student graduation status
- Provide ranking and reporting functionalities for administrative staff
- Ensure compliance with university policies by integrating them into the system's logic

Out of Scope:

- Handling of non-graduation-related academic processes such as course registration or student advising unrelated to graduation
- Management of post-graduation activities such as alumni relations

Benefits and Objectives:

- Improved operational efficiency by automating repetitive tasks
- Enhanced data accuracy and reduced human errors
- Simplified communication and transparent status tracking
- Compliance with institutional policies

1.3 Definitions, Acronyms, and Abbreviations

- AGMS: Automated Graduation Management System
- UBYS: University Information Management System
- GPA: Grade Point Average
- SRS: Software Requirements Specification

1.4 References

1. UBYS User Guide

A detailed guide explaining the functionalities of the University Information System (UBYS), including how to submit and track graduation applications.

2. Graduation Application Policies

The official policy document detailing eligibility criteria, required documents, deadlines, and procedures for applying for graduation.

(There is no certain source link for this reference but in the FAQ part of the student affairs' website, these policies are mentioned.)

3. Academic Regulations Handbook

A comprehensive handbook outlining academic requirements, credit completion standards, and procedures that govern the university's graduation process.

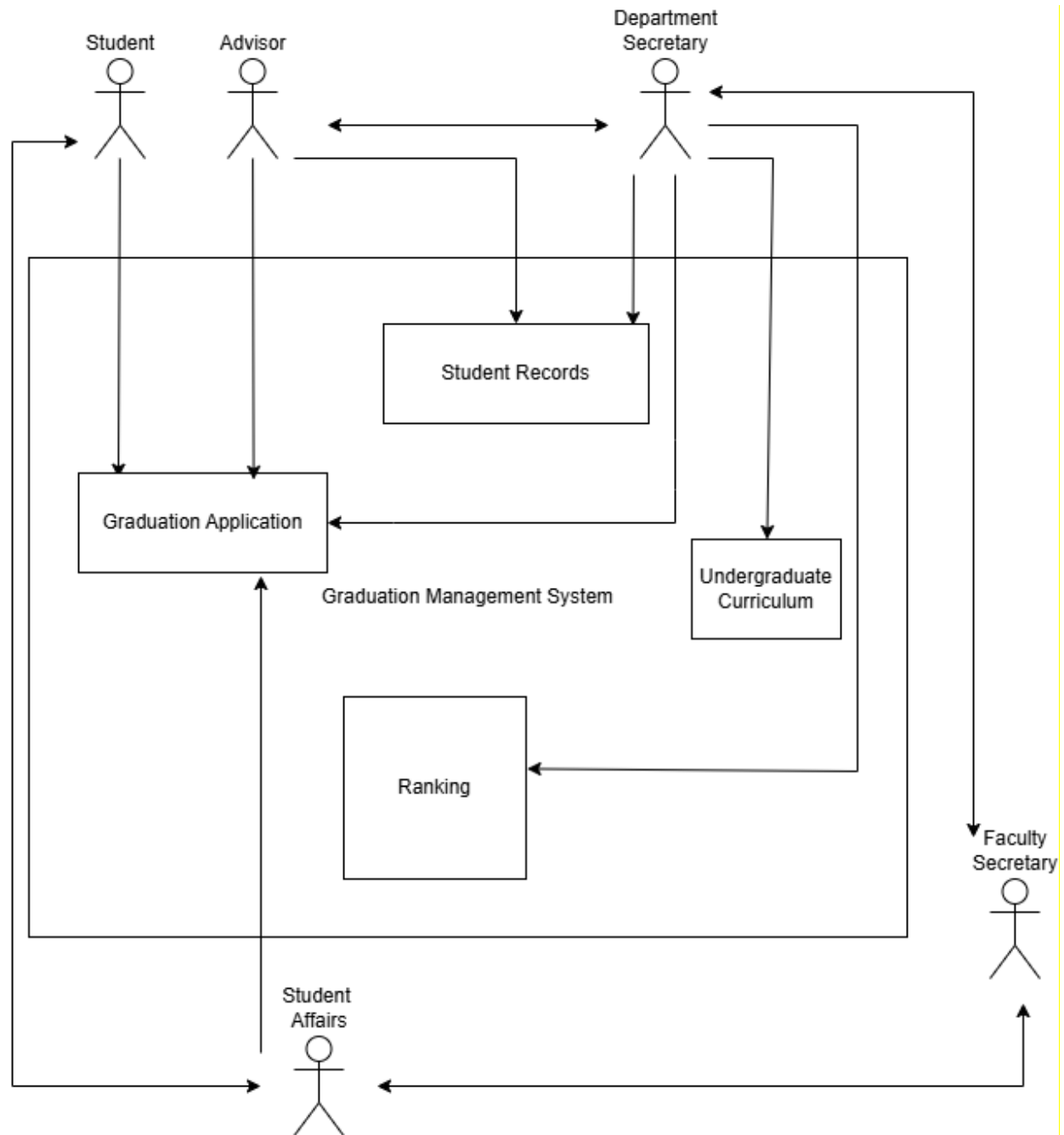
1.5 Overview

The following sections detail the requirements for AGMS:

- Overall Description (Section 2) provides general product context and background.
- Specific Requirements (Section 3) defines detailed use cases, performance criteria, and design constraints.

2. Overall Description

2.1 Product Perspective



The Automated Graduation Management System is a standalone application that integrates with UBYS for seamless data exchange.

- **System Interfaces:** The system relies on student academic records being exported from UBYS and imported into the graduation application system for processing.
- **User Interfaces:** Web-based portal accessible via desktop and mobile browsers.
- **Hardware Interfaces:** Server infrastructure supporting concurrent users.

- **Software Interfaces:** Integration with email servers for notifications.
- **Memory and Storage:** Secure storage of student data and backups.
- **Site Adaptation Requirements:** Hosted on a secure and reliable server infrastructure compliant with university IT policies.

2.2 Product Functions

The **Automated Graduation Management System (AGMS)** is designed to streamline and optimize graduation-related processes for students, advisors, and administrative staff. The main functions of the system include:

Student

1. **Submit Graduation Application:** Submit an application for graduation through the system.
2. **Check Graduation Status:** View the current status of their submitted graduation application.
3. **Resolve Application Issues:** Address issues related to the graduation application, such as missing documents or incorrect details.

Advisor

1. **Decide on Graduation Applications:** Approve or reject submitted graduation applications after review. (Backend yapıldı)
2. **Correct and Update Student Info:** Edit student information to ensure records are accurate. (furkan backend yapılacak)
3. **Communication with Secretary to Provide Necessary Documents:** Collaborate with the department secretary to share relevant documents for graduation processing. (çıkarılacak)

Department Secretary

1. **Import and Export Graduation Data:** Transfer student graduation data into or out of the system as needed.
2. **Submit Graduation Documents:** Provide finalized graduation documents to relevant authorities.
3. **Correct Graduation Issues:** Address errors or discrepancies in graduation records.
4. **Receive Approved Graduation Forms:** Collect approved graduation forms from advisors or department chairs.
5. **Rank Students by GPA:** Generate a ranking of graduating students based on their GPA.
6. **Prepare Cover Letter for Graduation Forms:** Draft and finalize cover letters to accompany graduation forms.

7. **Import Undergraduate Curriculum:** Add or update the department's undergraduate curriculum in the system.
8. **Submit Ranked List to Faculty Secretary:** Share the ranked list of students with the faculty secretary.
9. **Update Student Record:** Modify student records to reflect updates, such as approved graduation status.

Department Chair

1. **Sign Cover Letters and Send Them Back to the Department Secretary:** Review and sign prepared cover letters, returning them to the department secretary for further processing.

Faculty Secretary

1. **Inform Department Secretary About Not Approved Graduations:** Notify department secretaries of graduations that were not approved.
2. **Request Graduation Decision from Department Secretaries:** Collect graduation decisions from department secretaries for faculty-level processing.
3. **Send Graduation Decision to Faculties:** Submit finalized graduation decisions to the faculties.
4. **Check If Cover Letters Sent by Departments:** Monitor if cover letters have been sent by departments and follow up if necessary.
5. **Combine Graduating Students from Departments:** Consolidate a unified list of graduating students from all departments.
6. **Rank and Decide Top 3 of the Faculty:** Determine the top three students from the faculty based on GPA or other criteria.
7. **Send Transcripts, Cover Letters, Forms to Dean:** Forward required documents, including transcripts and cover letters, to the dean for approval.
8. **Send Documents Approved by Dean to Student Affairs:** Pass documents signed by the dean to the student affairs office.
9. **Inform Graduated Students:** Notify students of their successful graduation.

Student Affairs

1. **Rank the Students Among the University:** Rank graduating students at the university level based on performance or other criteria.
2. **Initiate and Request Graduation Decision from Faculties:** Prompt faculties to submit their graduation decisions.
3. **Check If Cover Letters Sent by Faculties:** Monitor and ensure faculties have sent required cover letters.

4. **Answer the Students' Questions:** Respond to inquiries from students regarding graduation processes.
5. **Update Student Record:** Modify records to reflect approved graduation statuses.

2.3 User Characteristics

The Automated Graduation Management System (AGMS) is designed for a diverse set of users, each with varying levels of technical expertise and responsibilities. Below are the user types and their general characteristics:

1. **Students:**
 - **Description:** Undergraduate students who are applying for graduation.
 - **Characteristics:**
 - Basic computer literacy.
 - Familiarity with using online systems such as UBYs.
 - Expected to interact with the system occasionally during the graduation process.
2. **Advisors (Instructors):**
 - **Description:** Faculty members responsible for reviewing and approving student applications.
 - **Characteristics:**
 - Moderate IT skills.
 - Familiarity with academic policies and curriculum requirements.
 - Regular interaction with the system during application review periods.
3. **Department Secretaries:**
 - **Description:** Administrative staff managing department-level graduation workflows.
 - **Characteristics:**
 - High familiarity with academic processes and record management.
 - Regular and frequent interaction with the system for data entry, list management, and communication with advisors and students.
4. **Department Chairs:**
 - **Description:** Heads of academic departments who oversee and approve graduation processes at the department level.
 - **Characteristics:**
 - Familiar with department operations and student academic records.
 - Moderate IT skills.
 - Periodic interaction with the system for approvals and document signing.
5. **Faculty Secretaries:**
 - **Description:** Administrative staff responsible for managing faculty-wide graduation workflows.
 - **Characteristics:**
 - Proficient in managing data across multiple departments.
 - Familiar with academic policies and reporting requirements.
 - Frequent interaction with the system during peak graduation periods for consolidating data and preparing reports.

6. **Dean's Office Staff:**
 - **Description:** Administrative staff working at the faculty level, handling approvals and coordinating with faculty secretaries.
 - **Characteristics:**
 - Proficient in document review and record management.
 - Familiar with high-level graduation policies.
 - Moderate interaction with the system for document approval and reporting.
7. **Student Affairs Staff:**
 - **Description:** Staff responsible for university-wide graduation processes, including final approvals and student communication.
 - **Characteristics:**
 - High-level IT and administrative skills.
 - Familiar with university-wide academic regulations and policies.
 - Regular interaction with the system for ranking students and updating records.
8. **Rectorate Staff:**
 - **Description:** Senior officials responsible for signing digital diplomas and ensuring compliance with university standards.
 - **Characteristics:**
 - Minimal interaction with the system, focused on digital approvals.
 - Require secure access and role-specific functionalities.

2.4 Constraints

The system must meet the following constraints:

1. **Regulatory Compliance:** Adhere to university data protection policies. Test by security audits and penetration testing.
2. **Language Support:** Provide interfaces in Turkish and English. Test by verifying translations across all features.
3. **Peak Load Handling:** Support 1,000+ concurrent users during peak periods with response times <3 seconds. Test through load testing.
4. **Availability:** Ensure 99.9% uptime during academic terms, excluding maintenance. Test via uptime monitoring logs.
5. **Data Backup:** Perform automated backups every 24 hours. Test by checking system logs and backup files.
6. **Security:** Implement secure authentication and role-based access. Test by simulating unauthorized attempts.
7. **Browser Compatibility:** Support major browsers and mobile devices. Test through compatibility testing.
8. **Audit Logging:** Record all user actions for audits. Test by verifying logs after various operations.

These constraints are measurable and can be validated through specific tests.

2.5 Assumptions and Dependencies

Reliable Internet Connectivity:

- Assumes consistent and stable internet access for all users.
- Impact: Unreliable connectivity may disrupt system access and user workflows.

User Computer Literacy:

- Assumes users have basic proficiency in using web-based applications.
- Impact: Insufficient skills may require additional user training or simplified interfaces.

Access to Student Records:

- Assumes that student academic records are periodically exported from the University Information Management System (UBYS) and imported into the graduation application system.
- Impact: Delays or errors in the export-import process from UBYS could disrupt essential functions like transcript validation and status tracking.

Availability of Student Academic Records:

- Assumes all required student records are complete and accurate in UBYS.
- Impact: Missing or incorrect records may lead to errors in validation and approval processes.

University Policies and Procedures Stability:

- Assumes no major changes to IZTECH's academic policies during development or deployment.
- Impact: Changes may necessitate updates to system workflows or logic.

2.6 Apportioning of Requirements

The following requirements have been identified as lower-priority features or enhancements that can be deferred to future versions of the system without affecting the core functionality of the current implementation:

Real-Time Notifications:

Description: Provide instant notifications via SMS or mobile app push notifications for status changes and issue updates.

Rationale: Current email-based notifications are sufficient for most users, and real-time alerts can be deferred until additional resources are available.

Future Plan: Evaluate after analyzing user feedback on existing notification mechanisms.

Multi-Language Support:

Description: Offer the user interface in multiple languages to accommodate a diverse student population, especially international students.

Rationale: The current version supports only English, which meets the majority of user needs. Adding multilingual support requires additional design and development.

Future Plan: Address in future versions based on demand and resource availability.

These requirements are postponed to balance project scope, development resources, and timeline, while ensuring the current version meets its primary objectives.

3. Specific Requirements

3.1 Use Cases

3.1.1 Use cases for user 1

3.1.1.1 Use Case 1 for Student

3.1.1.1.1 Use case template

Use Case ID:	1.1		
Use Case Name:	Submit Graduation Application		
Created By:	Mert Ali Kızılarıslan	Last Updated By:	Mert Ali Kızılarıslan
Date Created:	16/11/2024	Date Last Updated:	24/11/2024

Actor:	Student
Description:	The student submits a graduation application by completing a form, uploading required documents (e.g., course completion forms, disengagement form etc.) and confirming submission. The system validates the application and notifies relevant parties for further processing.

Preconditions:	<ol style="list-style-type: none"> 1. The student logs into the system. 2. The current date falls within the graduation application period.
Postconditions:	<ol style="list-style-type: none"> 1. The graduation application is submitted and securely stored in the system. 2. A confirmation message with a reference number is sent to the student. 3. The student's advisor is notified of the pending graduation application.
Priority:	It has high priority.
Frequency of Use:	Typically, once per student per degree program. However, students who cannot graduate on their initial application may reapply during subsequent graduation periods.
Normal Course of Events:	<ol style="list-style-type: none"> 1. Student selects "Apply for Graduation" option 2. System displays graduation application form 3. Student fills in required information 4. Student uploads any required documents (e.g., course completion forms, disengagement form etc.) 5. System validates all required fields 6. Student confirms submission 7. System saves application and generates confirmation number 8. System notifies the advisor about the pending application for review.
Alternative Courses:	<p>1.1.AC.1 - Student saves application as draft:</p> <ol style="list-style-type: none"> 1. Student saves partially completed application 2. System stores draft for later completion 3. Student can return to complete application within deadline
Exceptions:	<p>1.1.EX.1 - System Failure or Technical Issue:</p> <ol style="list-style-type: none"> 1. Condition: The system encounters a technical problem, such as a server error, data corruption, or unexpected behavior. 2. System Response:

	<ol style="list-style-type: none"> Displays a generic error message: <i>"An unexpected error occurred while processing your application. Please try again later."</i> Logs the error details in the system error logs for troubleshooting. Provides the student with a support contact (e.g., IT Helpdesk or technical support team) for assistance. If possible, retains any partially completed data to allow the student to resume the application process after the issue is resolved.
Includes:	
Special Requirements:	<ol style="list-style-type: none"> Form must be accessible on mobile devices Real-time field validation Support document upload up to 10MB per file The system must support multiple file formats for document uploads (PDF, DOCX, etc.)
Assumptions:	<ol style="list-style-type: none"> The student has a reliable internet connection. The required documents are available in digital format. The student's academic record is accurate and up-to-date.
Notes and Issues:	<ol style="list-style-type: none"> Define specific document requirements for international students. Clarify the escalation process for technical issues during the application process.

3.1.1.1.2 User Interface

[Home](#)[Link 1](#)[Link 2](#)[Link 3](#)[Profile](#)

Graduation Application Form

[Apply for Graduation](#)

Full Name

Student ID

Email Address

Phone Number

Date of Birth

- Personal Information
- Program Information
- Document Upload

[Submit](#)

3.1.1.1.3 Performance/Software Quality Requirements

- Performance:
 - Form submission processing < 3 seconds
 - Document upload processing < 5 seconds per file
- Reliability:
 - Automatic save of form data every 2 minutes
- Security:
 - Session timeout after 30 minutes

3.1.1.2 Use Case 2 for Student

3.1.1.2.1 Use case template

Use Case ID:	1.2		
Use Case Name:	Check Graduation Status		
Created By:	Mert Ali Kızılarıslan	Last Updated By:	Mert Ali Kızılarıslan
Date Created:	16/11/2024	Date Last Updated:	24/11/2024

Actor:	Student
Description:	Student monitors the progress of their graduation application through various approval stages and views any pending actions or issues that require attention.
Preconditions:	<ol style="list-style-type: none">1. The student has submitted a graduation application.2. The student's application is available in the system database.
Postconditions:	<ol style="list-style-type: none">1. The student successfully views the current status of their graduation application.2. The system logs a timestamp of the status check.3. Any new notifications related to the application are marked as read.
Priority:	It has medium priority.
Frequency of Use:	Weekly or as needed.
Normal Course of Events:	<ol style="list-style-type: none">1. The system displays the "Graduation Status" section.2. The student clicks the section to view progress.3. The system displays the current application status.

	<ol style="list-style-type: none"> 4. The student can navigate to view details about pending tasks. 5. System displays contact information for relevant department
Alternative Courses:	<p>1.2.AC.1 - Application Rejected:</p> <ul style="list-style-type: none"> ● Condition: The system detects that the student's graduation application has been rejected by one or more approving parties. ● Alternative Sequence of Events: <ol style="list-style-type: none"> 1. The student accesses the "Graduation Status" section to view their application. 2. The system displays a rejection message, including: <ul style="list-style-type: none"> ■ The reasons for rejection. ■ Specific tasks or corrections required to address the rejection. ■ Relevant deadlines for resubmission or appeal. 3. The student is guided to a separate interface to take corrective actions, such as updating information or contacting the appropriate department for assistance.
Exceptions:	<p>1.2.EX.1 - Failure to Retrieve Application Status:</p> <ul style="list-style-type: none"> ● Condition: The system fails to fetch the student's graduation application status due to technical issues, such as a server error or connection problem. ● System Response: <ol style="list-style-type: none"> 1. Displays an error message: <i>"We encountered an issue while retrieving your graduation application status. Please try again later."</i> 2. Provides a link to technical support or a troubleshooting guide. 3. Logs the error for system administrators to review. <p>1.2.EX.2 - System Maintenance or Update in Progress:</p>

	<ul style="list-style-type: none"> Condition: The system status is unavailable because it is undergoing maintenance or updates. System Response: <ol style="list-style-type: none"> Displays a notification message: <i>"The graduation status system is currently undergoing maintenance. Last updated on: [timestamp]."</i>
Includes:	
Special Requirements:	<ol style="list-style-type: none"> The system must provide real-time updates for any status changes.
Assumptions:	<ol style="list-style-type: none"> Application statuses are synced across departments.
Notes and Issues:	<ol style="list-style-type: none"> Clarify how the system handles conflicting or delayed updates from multiple departments.

3.1.1.2.2 User Interface

[Home](#)[Link 1](#)[Link 2](#)[Link 3](#)[Profile](#)

Graduation Status



Contact Information

▼ Advisor

- Name: Dr. Alice Johnson
- Email: alice.johnson@university.edu
- Phone: +1 (123) 456-7890
- Office Hours: Mon/Wed 2:00 PM - 4:00 PM
- Office Location: Room 301, Building A

- > Department Secretary
- > Department Chair
- > Faculty Secretary
- > Dean
- > Student Affairs
- > Rectorate

3.1.1.2.3 Performance/Software Quality Requirements

1. Response Time:
 - Notification delivery < 1 minutes
2. Reliability:
 - Update frequency: Every 5 minutes
3. Scalability:
 - Handle status checks every minute

3.1.1.3 Use Case 3 for Student

3.1.1.3.1 Use case template

Use Case ID:	1.3		
Use Case Name:	Resolve Application Issues		
Created By:	Mert Ali Kızılarıslan	Last Updated By:	Mert Ali Kızılarıslan
Date Created:	16/11/2024	Date Last Updated:	24/11/2024

Actor:	Student
Description:	Student addresses and resolves issues identified during the graduation application review process, including responding to department requests, updating information, and submitting additional documentation.
Preconditions:	<ol style="list-style-type: none">1. Student has an active graduation application2. Issues have been reported by reviewers3. Student has received notification of issues
Postconditions:	<ol style="list-style-type: none">1. All flagged issues are addressed and marked as resolved.2. Required updates or documents are submitted.

	<ol style="list-style-type: none"> 3. The system updates the application status to reflect the resolution. 4. Reviewers are notified of the resolution.
Priority:	It has high priority.
Frequency of Use:	As needed during application process
Normal Course of Events:	<ol style="list-style-type: none"> 1. Student receives issue notification 2. Student accesses "Application Issues" section 3. System displays list of pending issues with: <ul style="list-style-type: none"> • Issue description • Required actions • Deadline for resolution • Department contact information 4. Student selects issue to resolve 5. System provides appropriate resolution interface 6. Student submits required corrections/documents 7. System validates submissions 8. Student receives confirmation of submission 9. System notifies relevant department 10. Issue status updates to "Pending Review"
Alternative Courses:	<p>1.3.AC.1 - Contact Support for Assistance:</p> <ul style="list-style-type: none"> • Condition: The student cannot resolve the flagged issue independently due to unclear instructions or missing information. • Steps: <ol style="list-style-type: none"> 1. The student selects the "Contact Support" option in the "Application Issues" section. 2. The system displays support contact details and a messaging interface. 3. The student submits a query or request for assistance, including any relevant attachments or explanations. 4. The system sends the query to the support team and displays a confirmation message. 5. Support responds with guidance, enabling the student to proceed with resolving the issue.
Exceptions:	1.3.EX.2 - Unanticipated System Error:

	<ul style="list-style-type: none"> • Condition: A system error occurs during the resolution submission process. • Response: <ol style="list-style-type: none"> 1. Displays a general error message: <i>"An error occurred while processing your submission. Please try again later."</i> 2. Logs the error for system administrators to review. 3. Provides options to contact technical support for further assistance. 4. Maintains the issue status as unresolved.
Includes:	
Special Requirements:	<ol style="list-style-type: none"> 1. Notifications must be sent to all relevant stakeholders upon issue resolution. 2. The system must validate document formats and completeness before submission. 3. Real-time updates for issue statuses must be visible to the student and reviewers.
Assumptions:	<ol style="list-style-type: none"> 1. The flagged issues are actionable within the system. 2. All flagged issues are documented clearly with required actions specified.
Notes and Issues:	<ol style="list-style-type: none"> 1. Provide detailed instructions or FAQs to assist students in resolving common issues.

3.1.1.3.2 User Interface

[Home](#)[Link 1](#)[Link 2](#)[Link 3](#)[Profile](#)

Graduation Application Issues

! #Issue 1

Incorrect Program Information

- **Description:** Your declared program information does not match the department's records.
- **Required Action:** Update your program information in the system.

🕒 **Deadline for Resolution:** 27/11/2024



Department Contact:

- Name: Prof. Michael Smith
- Email: dept.chair@university.edu
- Phone: +1 (123) 456-7892

[Select](#)

! #Issue 2

Missing Required Credits

- **Description:** You have not completed the required number of credits for graduation.
- **Required Action:** Contact your advisor to review your academic plan.

🕒 **Deadline for Resolution:** 30/11/2024



Department Contact:

- Name: Dr. Alice Johnson
- Email: alice.johnson@university.edu
- Phone: +1 (123) 456-7890

[Select](#)

3.1.1.3.3 Performance/Software Quality Requirements

1. Reliability:

- Error Recovery:

In the event of a system error (e.g., failed file upload or validation), the system must recover and notify the student within 5 seconds while preserving their progress.

1. Usability:

- Clear Feedback:

The system must display real-time visual feedback (e.g., progress indicators, success/error messages) for actions like uploads, validation, and submissions within 500 milliseconds of the action.

1. Security:

- Access Control:

Only students with authenticated accounts tied to an active graduation application should access the "Application Issues" section.

- Error Privacy:

Error messages must not expose sensitive system or user information. Instead, provide specific guidance for resolving the issue (e.g., "File format not supported. Please upload PDF, DOCX, or JPEG.").

3.1.2 Use Cases for Advisor

3.1.2.1 Approve graduation application

3.1.2.1.1 Use Case Template

Use Case ID:	2.1		
Use Case Name:	Decide on graduation applications		
Created By:	Furkan Satır	Last Updated By:	Furkan Satır
Date Created:	18/11/2024	Date Last Updated:	18/11/2024

Actor:	Advisor
Description:	The advisor reviews a student's graduation application to ensure all academic and administrative requirements are met. If all conditions are satisfied, the advisor approves the application for further processing. If it is not, the advisor disapproves of the application.
Preconditions:	<ol style="list-style-type: none"> 1. The student must have submitted a graduation application through the system. 2. All academic records and grades must be finalized in the system. 3. The advisor has access to the graduation application and related data.
Postconditions:	<ol style="list-style-type: none"> 1. The graduation application is marked as approved in the system. 2. The student is notified of the application's approval.
Priority:	High
Frequency of Use:	At the end of every academic term or upon a student's request for a graduation check.
Normal Course of Events:	<ol style="list-style-type: none"> 1. Advisor accesses "Pending Graduation Reviews" dashboard. 2. System displays a list of submitted applications requiring review. 3. Advisor selects a student application to review. 4. System displays: <ul style="list-style-type: none"> • Student's complete transcript. • Program requirement checklist. • GPA calculations by category. • Outstanding requirements. • Previous academic warnings/issues. 5. Advisor reviews each requirement: <ul style="list-style-type: none"> • Core course completion. • Elective requirements. • Credit hour totals. • GPA thresholds.

	<ul style="list-style-type: none"> • Internship/practical requirements. <ol style="list-style-type: none"> 6. Advisor verifies all academic and administrative requirements are met. 7. Advisor approves the graduation application. 8. System updates the application status to "Approved". 9. System generates a notification to the student indicating the application was approved. 10. System forwards approval details to the department secretary for further processing.
Alternative Courses:	<ul style="list-style-type: none"> • 3.1.2.1.AC.1 - Rejection of incomplete application: If the application is incomplete, the advisor rejects or returns it with feedback for the student. • 3.1.2.1.AC.2 - Delegation of application review: The advisor can delegate application review to another authorized staff member.
Exceptions:	<ul style="list-style-type: none"> • 3.1.2.1.EX.1 - Missing or incorrect academic records: The system contains missing or incorrect academic records, preventing the advisor from accurately reviewing the application. The system notifies advisor about this issue. • 3.1.2.1.EX.2 - Technical issues: Technical issues occur while accessing or processing the application, delaying the review process. The system shows an error message that the process failed.
Includes:	
Special Requirements:	<ol style="list-style-type: none"> 1. Secure access to the graduation application system. 2. Tools to track and record application status changes.
Assumptions:	<ol style="list-style-type: none"> 1. The student's academic records are accurate and complete. 2. The system notifies the student automatically upon application approval or rejection.
Notes and Issues:	<ol style="list-style-type: none"> 1. Ensure timely review of applications to avoid delays in graduation processing.

2. Implement safeguards to prevent unauthorized approval or rejection of applications.

3.1.2.1.2 User Interface

logo

[Home](#) [Graduation List](#)

Profile









Graduation List

Import New List

Export As

Q Search

Short by: GPA

Student ID	Student Name	GPA	Email	Semester	Status
 STU 001	Name 1	3.77	a@b.com	Semester 8	<div>Ready</div>
 STU 002	Name 2	3.59	a@b.com	Semester 8	<div>Ready</div>
 STU 003	Name 3	3.5	a@b.com	Semester 8	<div>Ready</div>
 STU 004	Name 4	3.14	a@b.com	Semester 8	<div>Ready</div>
 STU 005	Name 5	2.67	a@b.com	Semester 8	<div>Not Ready</div>
 STU 006	Name 6	2.67	a@b.com	Semester 8	<div>Ready</div>
 STU 007	Name 7	2.24	a@b.com	Semester 8	<div>Ready</div>
 STU 008	Name 8	2.21	a@b.com	Semester 8	<div>Ready</div>

Showing data 1 to 8 of 3K entries

<

1

2

3

4

...

40

>

- After getting notification, advisor can access user details with edit button (located in the left of the Student IDs)
- After that a model pop up appears and he/she can see the required details that he/she has access.



Name Surname's information

It will take a couple of minutes to change profile settings and confirm.

General Information

Transcript

Internship information

Additional Informations



xth Grade yth Semester

	Lesson Name	Grade	Credit	AKTS	Status
	Lesson name 2	DD	3	5	Completed
	Lesson name 3	AA	4	6	Completed
	Lesson name 4	BC	4	6	Completed
	Lesson name 5	BA	3	5	Completed
	Lesson name 6	CD	5	7	Completed

Cancel

Save Changes

Confirm Transcript

Reject Transcript

- After clicking the edit button, advisor can see the information about student.
- After the checkup by advisor, he/she can confirm the transcript or can reject and student is informed about this decision.

3.1.2.1.3 Performance/Software Quality Requirements

Response Times

- Decide operations: < 1 minute transcript decision is done after the decide on transcript is done

Data Accuracy

- Field compliance: 100% adherence to templates.
- Error detection: Complete and user-notified.
- Exported data: Matches system database with no omissions.

Security

- Permission: Only an advisor can decide on the transcript

3.1.2.2 Correct and update student info

3.1.2.2.1 Use Case Template


Use Case ID:	2.2		
Use Case Name:	Correct and update student info		
Created By:	Furkan Satır	Last Updated By:	Furkan Satır
Date Created:	18/11/2024	Date Last Updated:	18/11/2024

Actor:	Advisor
Description:	The advisor reviews and updates a student's information in the system to ensure its accuracy. Corrections can include personal details, academic records, or other relevant data.
Preconditions:	<ol style="list-style-type: none">1. The student's current information is available in the system.2. The advisor has the necessary permissions to update student data.3. The student has requested corrections or the advisor identified discrepancies.
Postconditions:	<ol style="list-style-type: none">1. The student's information is updated and saved in the system.2. The system logs the changes for future reference.3. The student is notified of the updates.
Priority:	Medium

Frequency of Use:	As needed, when discrepancies are identified or correction requests are made by the student.
Normal Course of Events:	<ol style="list-style-type: none"> 1. Advisor accesses the "Student Information Management" dashboard. 2. System displays a list of students or a search option for locating a specific student. 3. Advisor searches for and selects the student whose information needs updating. 4. System displays the student's current records, including: <ul style="list-style-type: none"> • Personal details (e.g., name, date of birth, contact information). • Academic records (e.g., completed courses, grades, GPA). • Other relevant data (e.g., enrollment status, program details). 5. Advisor identifies incorrect or outdated information. 6. Advisor updates the necessary fields, such as: <ul style="list-style-type: none"> • Personal details (e.g., correcting a misspelled name or updating contact info). • Academic records (e.g., correcting a grade entry or updating course status). • Administrative details (e.g., correcting enrollment year or major). 7. Advisor enters a reason or note for the correction, if required. 8. Advisor submits the updates. 9. System validates the updates and saves the changes. 10. System generates a notification to the student, summarizing the updates. 11. System logs the changes for audit and tracking purposes.
Alternative Courses:	<ul style="list-style-type: none"> • 3.1.2.2.AC.1 - Forwarding unresolved requests: If the advisor cannot resolve the issue, the correction request is forwarded to the department secretary. • 3.1.2.2.AC.2 - Additional documentation request: The student provides additional documentation for verification, if needed.
Exceptions:	<ul style="list-style-type: none"> • 3.1.2.2.EX.1 - Insufficient or missing data: The system contains insufficient or missing data,

	<p>preventing the advisor from making the corrections. The system notifies advisor about this issue.</p> <ul style="list-style-type: none"> ● 3.1.2.2.EX.2 - Technical issues: Technical issues occur while updating the information in the system, delaying the correction process. The system shows an error message that if the process failed. The system retries automatically up to three times. A manual retry option is also provided. ● 3.1.2.2.EX.3 - System Downtime: ● If the system is under maintenance, a message is displayed indicating when the feature will be available. The system shows an error message that if the process failed.
Includes:	
Special Requirements:	<ol style="list-style-type: none"> 1. Secure authentication for access to sensitive student data. 2. Validation tools to ensure updated information is accurate and complete.
Assumptions:	<ol style="list-style-type: none"> 1. The student has provided accurate information for corrections. 2. The system automatically logs all updates for audit purposes.
Notes and Issues:	

3.1.2.2.2 User Interface



Name Surname's information

It will take a couple of minutes to change profile settings and confirm.






General Information

Transcript

Intership information

Additional Informations

xth Grade yth Semester

	Lesson Name	Grade	Credit	AKTS	Status
	Lesson name 2	DD	3	5	Completed
	Lesson name 3	AA	4	6	Completed
	Lesson name 4	BC	4	6	Completed
	Lesson name 5	BA	3	5	Completed
	Lesson name 6	CD	5	7	Completed
	<input type="text" value="Lesson name 7"/>	<input type="text" value="FD"/>	<input type="text" value="6"/>	<input type="text" value="8"/>	<input type="text" value="Not Completed"/>

Cancel

Save Changes

Confirm Transcript

Reject Transcript

- After clicking the edit button next to the relevant lesson name, advisor can edit the information about student.
- When advisor edits something, student is informed about this edit.

3.1.2.2.3 Performance/Software Quality Requirements

Response Times

- Edit operations: < 1 minute edit is saved after the edit is done.

Data Accuracy

- Field compliance: 100% adherence to templates.
- Error detection: Complete and user-notified.
- Exported data: Matches system database with no omissions.

Security

- Permission: Only an advisor can do this edit.

3.1.2.3 Communication with faculty secretary to give documents

3.1.2.3.1 Use Case Template

Use Case ID:	2.3		
Use Case Name:	Communication with faculty secretary to give documents		
Created By:	Furkan Satır	Last Updated By:	Furkan Satır
Date Created:	18/11/2024	Date Last Updated:	18/11/2024

Actor:	Advisor
Description:	The advisor communicates with the faculty secretary to request or provide necessary documents related to a student's academic or administrative processes.
Preconditions:	<ol style="list-style-type: none">1. The document(s) required must be identified and specified by the advisor or faculty secretary.2. Both the advisor and faculty secretary have access to the communication system or process.3. The advisor has all relevant student or faculty information to request or provide the document.
Postconditions:	<ol style="list-style-type: none">4. The required document is successfully provided to the faculty secretary.5. The system records the document transfer or communication for future reference.6. The document is used by the faculty secretary to complete the necessary academic or administrative process.


Priority:	Medium
Frequency of Use:	As needed, when a document is required for administrative or academic purposes.
Normal Course of Events:	<ol style="list-style-type: none"> 1. Advisor identifies the need to provide documents related to a student's academic or administrative processes. 2. Advisor accesses the "Faculty Communication" or "Document Management" dashboard in the system. 3. Advisor selects the relevant student and associated case from the system. 4. System displays: <ul style="list-style-type: none"> • Student's academic and administrative details. • List of documents associated with the student (e.g., transcripts, forms, approvals). 5. Advisor selects the documents to be sent to the faculty secretary. 6. Advisor adds a note or message to specify the purpose of the communication (e.g., "For graduation processing" or "For academic records correction"). 7. Advisor submits the communication request with attached documents. 8. System validates the documents and advisor sends the request to the faculty secretary via UBYS. 9. System generates a confirmation for the advisor indicating successful transmission of the documents. 10. Faculty secretary receives the documents and a notification of the advisor's request. 11. Faculty secretary acknowledges receipt of the documents, within the system.
Alternative Courses:	<ul style="list-style-type: none"> • 3.1.2.3.AC.1 - Forwarding unresolved requests: If the advisor cannot resolve the issue, the correction request is forwarded to the department secretary. • 3.1.2.3.AC.2 - Additional documentation request: The student provides additional documentation for verification, if needed.
Exceptions:	<ul style="list-style-type: none"> • 3.1.2.3.EX.1 - Insufficient or missing data: The system contains insufficient or missing data, preventing the advisor from making the corrections. The system notifies advisor about this issue.

	<ul style="list-style-type: none"> ● 3.1.2.3.EX.2 - Technical issues: Technical issues occur while updating the information in the system, delaying the correction process. The system shows an error message that if the process failed. The system retries automatically up to three times. A manual retry option is also provided. The system shows an error message that if the process failed. ● 3.1.2.3.EX.3 - File Format Errors: If the uploaded file is not in the required format, the system provides an error message and guidelines for correction. ● 3.1.2.3.EX.4 - System Downtime: If the system is under maintenance, a message is displayed indicating when the feature will be available.
Includes:	
Special Requirements:	<ol style="list-style-type: none"> 1. Secure and reliable communication channels to ensure the confidentiality of documents. 2. Clear identification of the required document and its purpose.
Assumptions:	<ol style="list-style-type: none"> 1. Both the advisor and faculty secretary are aware of the required documents. 2. The advisor has access to the documents or knows how to obtain them.
Notes and Issues:	<ol style="list-style-type: none"> 1. Ensure that the document complies with all relevant regulations before sending.

3.1.2.3.2 User Interface


logo

[Home](#) [Graduation List](#) [Send Docs](#) [Profile](#)





Upload Required Documents
Choose a file or drag & drop it here
PDF, CSV, XLSX, and DOCX formats, up to 100MB

Browse File



Student's documents

25 MB of 35 MB •  Uploading...



Submit

- After clicking on the Submit docs tab, the advisor sees a page that can upload required documents. After uploading them they can submit the docs with clicking the submit button to the faculty secretary.
- Note : The document for required documents can be accessible for advisor in his/her home page

3.1.2.3.3 Performance/Software Quality Requirements

Response Times

- Import/Export operations: < 15 seconds for files up to 100 MB.

Data Accuracy

- Field compliance: 100% adherence to templates.
- Error detection: Complete and user-notified.
- Exported data: Matches system database with no omissions.

Security

- Audit Trail: Tracks imports, exports, and edits.

3.1.3 Use Cases for Department Secretary

3.1.3.1 Manage Graduation List

3.1.3.1.1 Use Case Template

Use Case ID:	3.1		
Use Case Name:	Manage Graduation List		
Created By:	Yiğit Kılınçer	Last Updated By:	Yiğit Kılınçer
Date Created:	18/11/2024	Date Last Updated:	23/11/2024

Actor:	Department Secretary
Description:	The department secretary manages the graduation list by importing and exporting student data to ensure accuracy in graduation records. This process includes uploading CSV files, verifying data, and exporting final lists for review.
Preconditions:	<ol style="list-style-type: none">1. Graduation list format guidelines are predefined and can be accessible from the department secretary's home page.2. The system supports import/export functionalities for CSV or other formats.
Postconditions:	<ol style="list-style-type: none">1. Graduation list data is accurately updated in the system.2. Exported lists for external processing.

	3. Logs of the operation are maintained for audit purposes.
Priority:	It has high priority.
Frequency of Use:	Once during graduation periods
Normal Course of Events:	<p>1.The secretary navigates to the “Graduation List” module.</p> <p>2.For Importing:</p> <ul style="list-style-type: none"> • Selects “Import New List.” • Downloads the system-provided CSV template from home page, if required. • Uploads a completed CSV file. • The system validates the data: • Detects any errors and provides a summary. • Displays a list of the data for check and confirmation processes further. • Secretary confirms and submits the data. • The system updates the database and generates a success notification. <p>3.For Exporting:</p> <ul style="list-style-type: none"> • Selects “Export List.” • Specifies the format (e.g., CSV, Excel). • Confirms export options. • The system generates and downloads the file or sends it via email. <p>4.Logs of the import/export operation are saved for audit purposes.</p>
Alternative Courses:	<p>3.1.3.1.AC.1. Import Validation Errors:</p> <ul style="list-style-type: none"> • If errors are detected during the upload, the system displays a detailed report. • The secretary can either correct errors manually in the system or re-upload a corrected file. <p>3.1.3.1.AC.2. Interrupted Export:</p>

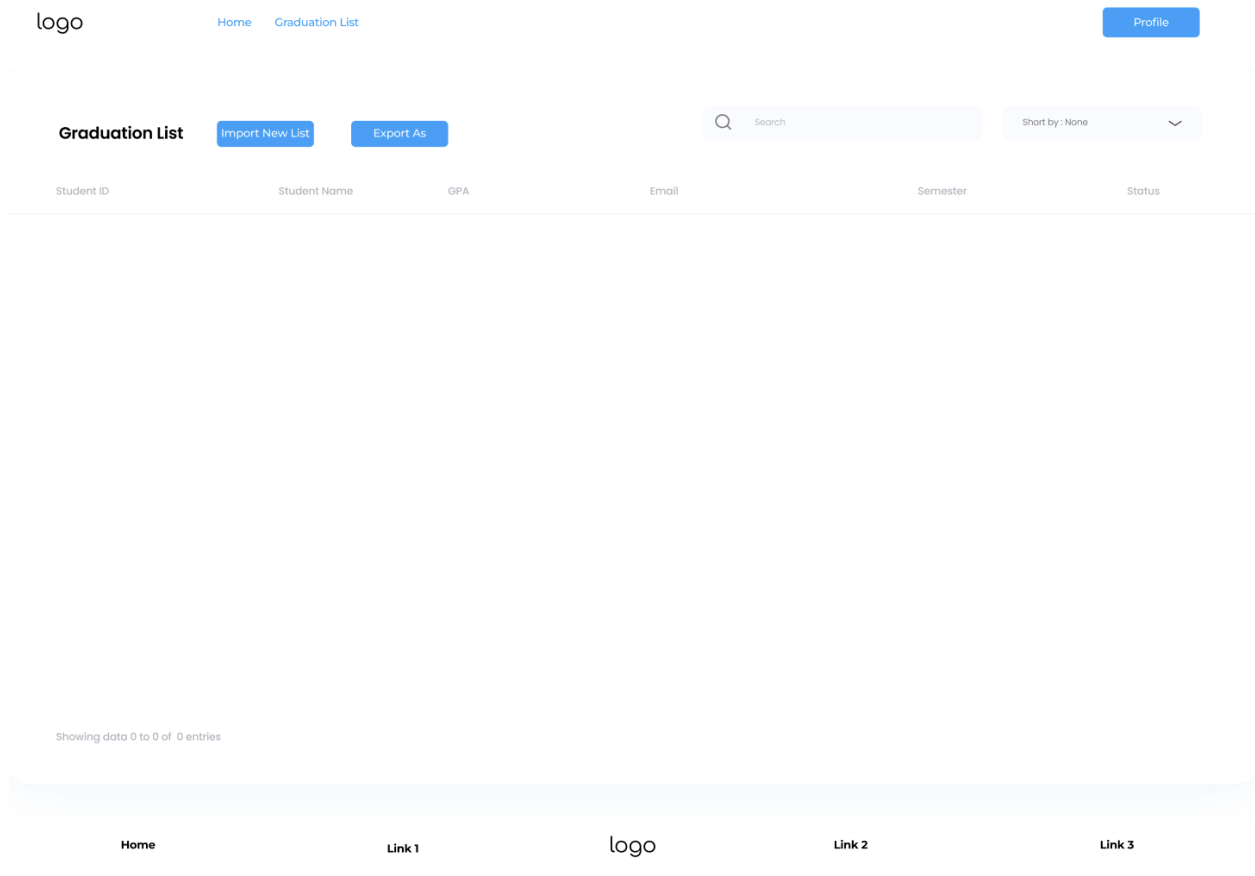
	<ul style="list-style-type: none"> If the export operation fails (e.g., network issues), the system retries automatically up to three times. A manual retry option is also provided.
Exceptions:	<p>3.1.3.1.EX.1. File Format Errors:</p> <ul style="list-style-type: none"> If the uploaded file is not in the required format, the system provides an error message and guidelines for correction. <p>3.1.3.1.EX.2. Data Mismatch:</p> <ul style="list-style-type: none"> If the imported data contains mismatched fields (e.g., unrecognized student IDs), the system highlights these and prevents submission. <p>3.1.3.1.EX.3. System Downtime:</p> <ul style="list-style-type: none"> If the system is under maintenance, an error message is displayed with the next available time for operation. <p>3.1.3.1.EX.4.. Permission Denied:</p> <ul style="list-style-type: none"> If an unauthorized user attempts to access the module, the system denies access and logs the attempt.
Includes:	<p>Use Case 3.1.2.5 Communication with faculty secretary to give documents</p> <p>Use Case 3.1.3.3 Correct graduation issues</p>
Special Requirements:	<p>1. Error Reporting: Clear and detailed error messages should be displayed for invalid data.</p> <p>2. Data Security: Ensure secure handling of sensitive student information during import/export operations.</p> <p>3. Usability: Provide a user-friendly interface for secretaries with minimal technical expertise</p>
Assumptions:	<p>1. The department secretary has a basic understanding of how to work with CSV and Excel files.</p> <p>2. The graduation criteria and data format are clearly defined and provided to the secretary.</p>

	3. The system's server can handle concurrent operations without significant performance degradation.
Notes and Issues:	<p>1. A feature to automate frequent imports/exports for large departments may be considered in future updates.</p> <p>2. Current error detection is limited to syntactical issues; future enhancements may include semantic checks (e.g., verifying student eligibility).</p> <p>3. Ensure regular backups of imported/exported data to avoid loss during unforeseen errors.</p>

3.1.3.1.2 User Interface

- After navigating to Graduation List tab department secretary sees this page

Before Import :



- Then department secretary clicks import new list button and imports csv and system shows students as a list provided below

After Import:

logo

[Home](#)[Graduation List](#)

Profile

Graduation List

Import New List

Export As

Q

Search

Short by: Student Name

Student ID	Student Name	GPA	Email	Semester	Status
STU 001	Name 1	3.14	a@b.com	Semester 8	Ready
STU 002	Name 2	3.59	a@b.com	Semester 8	Ready
STU 003	Name 3	2.24	a@b.com	Semester 8	Ready
STU 004	Name 4	2.03	a@b.com	Semester 8	Ready
STU 005	Name 5	3.77	a@b.com	Semester 8	Ready
STU 006	Name 6	3.5	a@b.com	Semester 8	Ready
STU 007	Name 7	2.21	a@b.com	Semester 8	Ready
STU 008	Name 8	2.67	a@b.com	Semester 8	Not Ready

Showing data 1 to 8 of 3k entries

<

1

2

3

4

...

40

>

Home

Link 1

logo

Link 2

Link 3

3.1.3.1.3 Performance/Software Quality Requirements

Response Times

- Import/Export operations: < 15 seconds for files up to 100 MB.

Data Accuracy

- Field compliance: 100% adherence to templates.
- Error detection: Complete and user-notified.
- Exported data: Matches system database with no omissions.

Security

- Audit Trail: Tracks imports, exports, and edits.

3.1.3.2 Submit Graduation Documents

3.1.3.2.1 Use Case Template

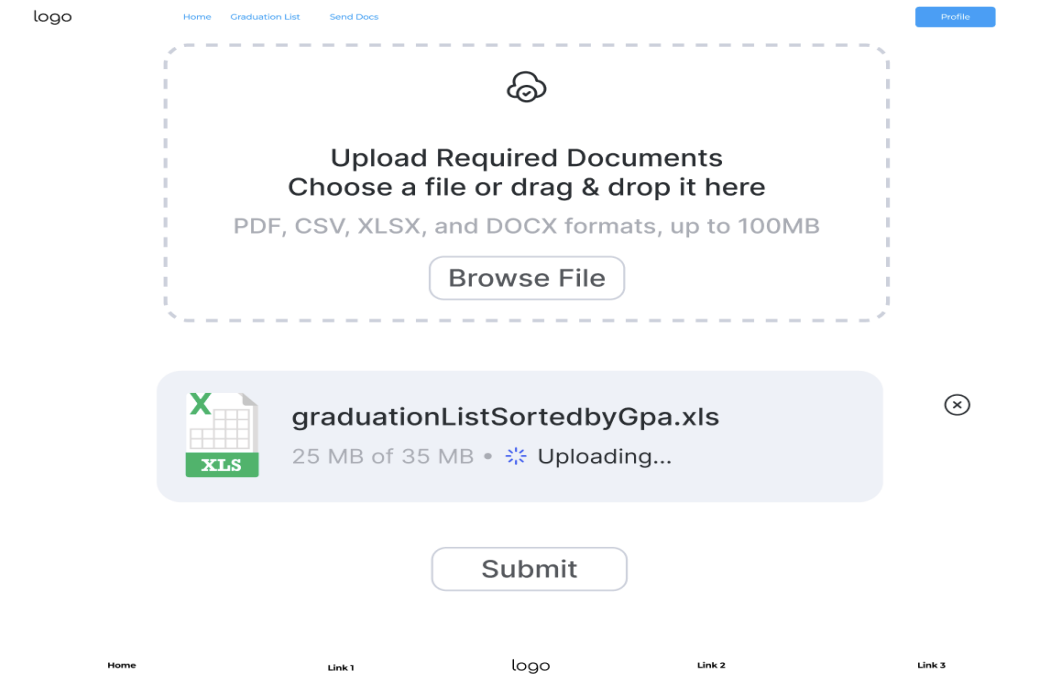
Use Case ID:	3.2		
Use Case Name:	Submit Graduation Documents		
Created By:	Yiğit Kılınçer	Last Updated By:	Yiğit Kılınçer
Date Created:	18/11/2024	Date Last Updated:	23/11/2024

Actor:	Department Secretary
Description:	The department secretary submits and manages final graduation documents, ensuring accurate and complete student graduation records for institutional processing and official documentation.
Preconditions:	1. All graduation documents are checked and ready to send. 2. Document submission guidelines are predefined.
Postconditions:	1. Graduation document submissions are accurate and complete. 2. Verified documents are saved in the system. 3. Records of submission are logged for audit purposes.
Priority:	It has high priority.
Frequency of Use:	Once during graduation periods
Normal Course of Events:	1.The secretary navigates to the “Send Docs” module.

	<p>2.Selects “Browse File” option or just drag and drop the files.</p> <p>3.Uploads required files according to guidelines.</p> <p>4.The system performs a verification check:</p> <ul style="list-style-type: none"> • Identifies missing or incorrect documents • Alerts issues for correction <p>5. The secretary resolves flagged issues and resubmits, if necessary.</p> <p>6. After verification, the secretary finalizes the submission.</p> <p>7. System confirms successful submission and logs.</p>
Alternative Courses:	None
Exceptions:	<p>3.1.3.2.EX.1. Incorrect File Format:</p> <ul style="list-style-type: none"> • If files are in unsupported formats, the system displays an error message and provides guidelines for correction. <p>3.1.3.2.EX.3. System Downtime:</p> <ul style="list-style-type: none"> • If the system is under maintenance, a message is displayed indicating when the feature will be available.
Includes:	<p>Use Case 3.1.2.5 Communication with faculty secretary to give documents</p> <p>Use Case 3.1.3.3 Correct graduation issues</p>
Special Requirements:	<p>1. File Format Support: The system must support common file formats (e.g., PDF, DOCX).</p> <p>2. Error Reporting: Clear and detailed messages for invalid or missing files.</p> <p>3. Data Security: Ensure secure handling and storage of sensitive student information.</p>

Assumptions:	<ol style="list-style-type: none">1. The department secretary is familiar with graduation document requirements.2. All documents follow a predefined naming and formatting standard.3. The system supports concurrent users without significant performance degradation.
Notes and Issues:	<ol style="list-style-type: none">1. Future iterations could include AI-based verification for document content compliance.2. Cloud based solutions like AWS S3 can be implemented due its fast, secure and storage friendly approach.3. Ensure regular backups of submitted documents to avoid loss during unforeseen issues.4. Explore automation for bulk document submission in large departments.

3.1.3.2.2 User Interface



- After clicking on the Submit docs tab, the department secretary sees a page that can upload required documents. After uploading them they can submit the docs with clicking the submit button to the relevant person.
- Note : The document for required documents can be accessible for department secretary in his/her home page
- Note : Home page is the default opening page for the Department secretary's interface. Then the Department secretary accesses the Graduation List page or Submit docs page by clicking on the tabs.

3.1.3.2.3 Performance/Software Quality Requirements

Response Times

- File upload and verification: < 10 seconds for up to 100 MV.

Data Accuracy

- Error detection: All issues flagged and user-notified.
- Submitted data: Matches uploaded files without discrepancies.

Security

- File sensitization against malicious updates.
- Use rate limiting to avoid attacks.

Scalability

- File size: Individual files up to 100 MB each.

3.1.3.3 Correct graduation issues

3.1.3.3.1 Use Case Template

Use Case ID:	3.3		
Use Case Name:	Correct graduation issues		
Created By:	Yiğit Kılınçer	Last Updated By:	Yiğit Kılınçer

Date Created:	19/11/2024	Date Last Updated:	23/11/2024
---------------	------------	--------------------	------------

Actor:	Department Secretary
Description:	This use case describes the process where the department secretary reviews, identifies, and corrects issues in student graduation records to ensure accurate graduation status and documentation.
Preconditions:	<ol style="list-style-type: none"> 1. Student graduation record exists in the system 2. Issue has been reported or identified in the graduation status 3. Secretary has necessary authorization to modify graduation records
Postconditions:	<ol style="list-style-type: none"> 1. Graduation record has been corrected 2. All changes are logged in the system 3. Relevant stakeholders are notified of the corrections 4. Updated documentation is generated if required
Priority:	It has high priority.
Frequency of Use:	As much as there are issues related to department secretaries role description in graduation periods.
Normal Course of Events:	<ol style="list-style-type: none"> 1. Secretary receives notification or identifies graduation issue 2. Secretary logs into the student information system 3. System displays the students list from 'Graduation List' module 4. Secretary clicks edit button for specific student record

	<p>5. System displays student graduation information</p> <p>6. Secretary reviews the graduation requirements and current status</p> <p>7. Secretary identifies the specific issues in the record</p> <p>8. Secretary makes necessary corrections to the graduation record</p> <p>9. System validates the changes</p> <p>10. Secretary adds notes explaining the corrections made</p> <p>11. Secretary saves the updated record</p> <p>12. System confirms successful update</p> <p>13. Secretary generates updated graduation documentation</p> <p>14. System notifies relevant parties of the changes</p>
Alternative Courses:	<p>3.1.3.3.AC.1 - Multiple Issues Found:</p> <ul style="list-style-type: none"> • After step 7, secretary discovers multiple related issues • Secretary documents all found issues • Secretary prioritizes corrections needed • Secretary processes corrections in order of priority • Returns to step 8 <p>3.1.3.3.1.AC.2 - Required Documentation Missing</p> <ul style="list-style-type: none"> • At step 6, secretary discovers missing documentation • Secretary contacts relevant departments/student for documentation • Secretary puts correction on hold pending documentation • Upon receiving documentation, returns to step 7
Exceptions:	<p>3.1.3.3.1.EX.1 - Invalid Correction Attempt</p> <ul style="list-style-type: none"> • System prevents invalid changes • System displays validation errors • Secretary reviews requirements and policies • Returns to step 7

Includes:	<p>Use Case 3.1.2.5 Communication with faculty secretary to give documents</p> <p>Use Case 3.1.3.2 Submit Graduation Documents</p>
Special Requirements:	<ol style="list-style-type: none"> 1. All changes must be tracked with timestamp and user ID 2. System must maintain audit trail of all corrections 3. System must enforce graduation requirement rules
Assumptions:	<ol style="list-style-type: none"> 1. Secretary has knowledge of graduation requirements 2. Required documentation formats are standardized 3. System maintains real-time data synchronization 4. Backup systems are in place for data security
Notes and Issues:	<p>Need to develop standardized correction procedures</p> <p>Consider implementing automated validation rules</p> <p>Review notification system for stakeholders</p> <p>TBD: Integration with digital signature system</p>

3.1.3.3.2 User Interface

- After getting notification, department secretary can access user details with edit button (located in the left of the Student IDs)
- After that a model pops up and he/she can edit the required details that he/she has access to.
- The pop up is the 3.1.2.2.2 User Interface 's 'General Information' tab which contains Student details and some of are editable by the department secretary and some of are locked (department secretary does not have access to edit it.).
- After edit has made there is a save button on the bottom right of the pop up that department secretary can save by clicking on it.

logo	Home	Graduation List			Profile
Graduation List		Import New List	Export As	Search	Short by: Student Name
Student ID	Student Name	GPA	Email	Semester	Status
STU 001	Name 1	3.14	a@b.com	Semester 8	Ready
STU 002	Name 2	3.59	a@b.com	Semester 8	Ready
STU 003	Name 3	2.24	a@b.com	Semester 8	Ready
STU 004	Name 4	2.03	a@b.com	Semester 8	Ready
STU 005	Name 5	3.77	a@b.com	Semester 8	Ready
STU 006	Name 6	3.5	a@b.com	Semester 8	Ready
STU 007	Name 7	2.21	a@b.com	Semester 8	Ready
STU 008	Name 8	2.67	a@b.com	Semester 8	Not Ready
Showing data 1 to 8 of 3K entries					1 2 3 4 ... 40
Home	Link 1	logo	Link 2	Link 3	

3.1.3.3.3 Performance/Software Quality Requirements

Security:

Encryption of sensitive data of the student

No other performance requirements in context of this use case.

3.1.3.4 Receive approved graduation forms

3.1.3.4.1 Use Case Template

Use Case ID:	3.4		
Use Case Name:	Receive approved graduation forms		
Created By:	Yiğit Kılınçer	Last Updated By:	Yiğit Kılınçer
Date Created:	19/11/2024	Date Last Updated:	23/11/2024

Actor:	Department Secretary
Description:	This use case describes the process of receiving, validating, and processing approved graduation forms from advisors and ensuring their proper documentation in the system.
Preconditions:	<ol style="list-style-type: none"> 1. Graduation forms have been approved by advisors 2. Digital or physical forms are available for processing 3. Student records are accessible in the system
Postconditions:	<ol style="list-style-type: none"> 1. All received forms are properly logged in the system 2. Forms are digitally archived 3. Confirmation receipts are generated 4. Student graduation status is updated 5. Faculty secretary is notified of form receipt
Priority:	It has high priority.
Frequency of Use:	Once per graduation period. (If there are no edge cases)
Normal Course of Events:	<ol style="list-style-type: none"> 1. Secretary receives approved graduation forms 2. Secretary logs into the form processing system 3. System displays the form from the 'Graduation List' module after clicking the relevant students edit button. 4. Secretary verifies form completeness and signatures 5. Secretary enters or scans form information into system 6. System validates entered information

	<p>7. Secretary attaches digital copies if physical forms</p> <p>8. Secretary assigns proper categorization and tags</p> <p>9. System updates student graduation status</p> <p>10. Secretary generates receipt confirmation then System sends notifications to relevant parties (faculty secretaries)</p> <p>11. System archives the form in digital repository</p>
Alternative Courses:	<p>3.1.3.4.AC.1 - Incomplete Form Reception</p> <ul style="list-style-type: none"> • During verification, secretary identifies missing elements • Secretary logs incomplete items • Secretary notifies relevant department/student • Process holds until completion • Returns to step 4 upon receiving complete information <p>3.1.3.4.AC.2 - Batch Processing</p> <ul style="list-style-type: none"> • Secretary receives multiple forms simultaneously • System enables batch scanning/processing • Secretary verifies each form individually • System processes forms in queue • Returns to step 8 for each form
Exceptions:	<p>3.1.3.4.EX.1 - Invalid Form Detection</p> <ul style="list-style-type: none"> • System flags invalid or unauthorized forms • Secretary investigates discrepancy • Notifies relevant authorities if necessary • Documents invalid submission attempt <p>3.1.3.4.EX.2 - System Technical Issues</p> <ul style="list-style-type: none"> • Secretary documents forms received during system downtime • Implements backup reception procedure which ensures all the files and required information is backed up according to the procedure which is provided to the department secretaries. • Processes backlog when system restores • Verifies all data once system is operational

Includes:	Use Case 3.1.3.10 Update student record
Special Requirements:	<ol style="list-style-type: none"> 1. Digital archiving capability 2. Secure document storage 3. Multiple format support (PDF, DOCX, scanned images) 4. Automated notification system
Assumptions:	<ol style="list-style-type: none"> 1. Forms follow standardized format 2. Digital scanning equipment is available 3. Network connectivity is reliable 4. Storage capacity is sufficient
Notes and Issues:	<p>Consider implementing OCR for physical forms</p> <p>Review storage capacity quarterly</p> <p>Develop backup reception procedures</p> <p>TBD: Integration with UBYS</p>

3.1.3.4.2 User Interface

- Department Secretary can access the received graduation form as a pdf file about relevant student with clicking the edit button.

logo

Home

Graduation List

Profile

Graduation List

Import New List

Export As

Q

Search

Short by: Student Name

Student ID	Student Name	GPA	Email	Semester	Status
STU 001	Name 1	3.14	a@b.com	Semester 8	Ready
STU 002	Name 2	3.59	a@b.com	Semester 8	Ready
STU 003	Name 3	2.24	a@b.com	Semester 8	Ready
STU 004	Name 4	2.03	a@b.com	Semester 8	Ready
STU 005	Name 5	3.77	a@b.com	Semester 8	Ready
STU 006	Name 6	3.5	a@b.com	Semester 8	Ready
STU 007	Name 7	2.21	a@b.com	Semester 8	Ready
STU 008	Name 8	2.67	a@b.com	Semester 8	Not Ready

Showing data 1 to 8 of 3K entries

<

1

2

3

4

...

40

>

Home

Link 1

logo

Link 2

Link 3

3.1.3.4.3 Performance/Software Quality Requirements

Data Management

Storage Capacity: Minimum 10TB (for all system)

Backup Frequency: Every 6 hours

File Size Support: Up to 10MB per form

Security Standards

Data Encryption: AES-256

3.1.3.5 Rank students by gpa

3.1.3.5.1 Use Case Template

Use Case ID:	3.5		
Use Case Name:	Rank students by gpa		
Created By:	Yiğit Kılınçer	Last Updated By:	Yiğit Kılınçer
Date Created:	19/11/2024	Date Last Updated:	23/11/2024









Actor:	Department Secretary
Description:	This use case describes the process of generating, viewing, and managing student rankings based on their Grade Point Averages (GPA) for graduation purposes and academic standing.
Preconditions:	<ol style="list-style-type: none">1. All student grades are properly recorded in the system2. GPA calculations are up-to-date3. Academic period is properly defined (semester/year)
Postconditions:	<ol style="list-style-type: none">1. Student rankings are generated and stored2. Ranking reports are available for authorized viewing3. System logs ranking generation activity4. Notifications sent to relevant stakeholders (Dean, Dean secretary) if required
Priority:	It has high priority.

Frequency of Use:	Once per graduation period. (If there are no edge cases)
Normal Course of Events:	<ol style="list-style-type: none"> 1. Secretary accesses the 'Graduation List' module 2. Secretary selects rank by GPA 3. System generates preliminary ranking list 4. Secretary verifies data accuracy 5. Secretary applies any necessary filters or adjustments (For example, if highlight for the top x students needed applies this filter. Or another example if a data needed for below a specific gpa this filter can be applicable.) 6. System finalizes student rankings 7. Secretary reviews final ranking list 8. Secretary generates required reports 9. System saves ranking data 10. Secretary distributes rankings as needed
Alternative Courses:	<p>3.1.3.5.AC.1 - Specific Group Ranking</p> <ul style="list-style-type: none"> ● Secretary selects specific student group ● System filters student data based on criteria ● Rankings generated for selected group only ● Returns to step 7
Exceptions:	<p>3.1.3.5.EX.1 - Incomplete Grade Data</p> <ul style="list-style-type: none"> ● System identifies students with incomplete grades ● Secretary marks affected records ● System excludes incomplete records from ranking ● Secretary notes exceptions in report <p>3.1.3.5.EX.2 - Tie Resolution</p> <ul style="list-style-type: none"> ● System identifies tied GPAs ● System applies secondary ranking criteria (graduation semester or special achievements) ● Secretary reviews tie-breaking results

	<ul style="list-style-type: none"> System updates rankings accordingly
Includes:	Use Case 3.1.2.5 Communication with faculty secretary to give documents
Special Requirements:	<ol style="list-style-type: none"> 1. Ability to handle decimal precision in GPA calculations 2. Support for different ranking methodologies 3. Historical ranking data preservation 4. Custom report generation capabilities
Assumptions:	<ol style="list-style-type: none"> 1. All GPAs are calculated using the same methodology 2. Student records are current and accurate 3. Ranking criteria are standardized 4. System has sufficient processing capacity
Notes and Issues:	<p>Consider implementing real-time ranking updates</p> <p>Review ranking methodology annually</p> <p>Develop automated verification procedures</p> <p>TBD: Integration with honors system</p>

3.1.3.5.2 User Interface

- Department secretary can sort graduation list by clicking the dropdown 'sort by' and selecting 'GPA'
- After solving graduation issues (which is shown in the status column), the secretary can export a list of ready for graduation students and submit it to the relevant person.

logo		Home	Graduation List	Profile	
Graduation List		Import New List	Export As	<div> <div>Q</div> <div>Search</div> </div>	<div> <div>Short by : GPA</div> <div>▼</div> </div>
Student ID	Student Name	GPA	Email	Semester	Status
 STU 001	Name 1	3.77	a@b.com	Semester 8	Ready
 STU 002	Name 2	3.59	a@b.com	Semester 8	Ready
 STU 003	Name 3	3.5	a@b.com	Semester 8	Ready
 STU 004	Name 4	3.14	a@b.com	Semester 8	Ready
 STU 005	Name 5	2.67	a@b.com	Semester 8	Not Ready
 STU 006	Name 6	2.67	a@b.com	Semester 8	Ready
 STU 007	Name 7	2.24	a@b.com	Semester 8	Ready
 STU 008	Name 8	2.21	a@b.com	Semester 8	Ready
Showing data 1 to 8 of 3K entries		<div> <div><</div> <div>1</div> <div>2</div> <div>3</div> <div>4</div> <div>...</div> <div>40</div> <div>></div> </div>			
Home	Link 1	logo	Link 2	Link 3	

3.1.3.5.3 Performance/Software Quality Requirements

1. Processing Performance

Ranking Calculation Time: < 5 seconds for 1000 students

Report Generation: < 10 seconds for standard reports

Data Refresh Rate: Real-time updates

2. Data Accuracy

GPA Precision: 2 decimal places (assumed data delivered in a format like x.yz)

3. System Capacity

Student Records: Support for 1,000 to 5,000 records

3.1.3.6 Prepare cover letter for graduation forms

3.1.3.6.1 Use Case Template

Use Case ID:	3.6		
Use Case Name:	Prepare cover letter for graduation forms		
Created By:	Şahin Ökdem	Last Updated By:	Şahin Ökdem
Date Created:	18/11/24	Date Last Updated:	23/11/24

Actor:	Department Secretary
Description:	The Department Secretary prepares a cover letter for graduation forms, summarizing the student's graduation application details and attaching relevant documentation for further approval by the academic advisors and the dean.
Precondition:	<ul style="list-style-type: none">• The Department Secretary must be logged into the system with appropriate permissions.• The student's graduation form must already be submitted and verified.
Postconditions:	<ul style="list-style-type: none">• The cover letter is generated and attached to the student's graduation form.• The prepared document is ready for review and approval by academic advisors or higher authorities.

Priority:	High
Frequency of Use:	Each time a student submits a graduation form
Normal Course of Events:	<ol style="list-style-type: none"> 1. The Department Secretary logs into the system. 2. The Secretary navigates to the "Graduation Management" section. 3. The Secretary selects the "Prepare Cover Letter" option for a specific graduation form. 4. The system auto-populates student details (e.g., name, department, GPA, and graduation requirements status). 5. The Secretary reviews the pre-filled details and adds any additional comments or notes. 6. The Secretary clicks "Generate Cover Letter." 7. The system generates the cover letter and attaches it to the corresponding graduation form. 8. The system displays a confirmation message indicating successful generation.
Alternative Courses:	<ul style="list-style-type: none"> ● 3.6.AC.1: If student details are incomplete or incorrect, the system prompts the Secretary to update the records before proceeding. ● 3.6.AC.2: If the system fails to generate the cover letter, it provides troubleshooting options (e.g., retry or contact support).
Exceptions:	<ul style="list-style-type: none"> ● 3.6.EX.1: Missing Information: If required student details are missing, the system alerts the Secretary to fill in the missing data. ● 3.6.EX.1: System Downtime: If the system is unavailable, the process cannot proceed, and manual preparation may be required.
Includes:	
Special Requirements:	<ul style="list-style-type: none"> ● The system should ensure consistency in formatting and structure of all generated cover letters.

	<ul style="list-style-type: none"> • The system should provide a preview option before finalizing the document.
Assumptions:	The Department Secretary is familiar with the required structure and content of the cover letter.
Notes and Issues:	<ul style="list-style-type: none"> • Consider adding a template customization feature for different departments or programs. • Ensure generated documents are compliant with official university guidelines.

3.1.3.6.2 User Interface

LOGO[Home](#)[Cover Letter](#)

Cover Letter

Student Information

290201032
Şahin Ökdem
sahinokdem@std.iyte.edu.tr

Department: Computer Engineering

GPA: 3.1

Completed Credits: 81

Missing Courses: None

Disciplinary Record Status: None

Graduation Status: Eligible

Graduation Application Date: 10/06/2026

Tuition Fee Status: Paid

Any Additional Notes:

GENERATE COVER LETTER

Graduation Management Interface:

- Select graduation form with search functionality by student name or ID.
- Auto-populated fields for student information.
- Editable fields for additional comments or notes.

3.1.3.6.3 Performance/Software Quality Requirements

Response Times:

- Auto-populate student details: ≤ 2 seconds.
- Generate cover letter: ≤ 3 seconds.

Data Accuracy:

- Auto-populated student details must match input records with 100% accuracy.

System Availability:

- Available during core hours (8AM-6PM) with 99% uptime.

Security:

- Only authorized users can generate cover letters.

Scalability:

- Handle up to 20 concurrent cover letter generations without slowing down.

3.1.3.7 Import the undergraduate curriculum**3.1.3.7.1 Use Case Template**

Use Case ID:	3.7		
Use Case Name:	Import the undergraduate curriculum		
Created By:	Şahin Ökdem	Last Updated By:	Şahin Ökdem
Date Created:	18/11/24	Date Last Updated:	23/11/24

Actor:	Department Secretary
Description:	The Department Secretary imports the undergraduate curriculum into the Graduation Management System to ensure accurate course and credit information for graduation requirement validation.
Preconditions:	<ul style="list-style-type: none"> • The Department Secretary must be logged into the system with the appropriate permissions. • The curriculum file must be prepared in the required format.(XLS, XLSX, CSV)
Postconditions:	<ul style="list-style-type: none"> • The undergraduate curriculum is successfully uploaded to the system. • The curriculum data is available for use in credit validation and other related processes.
Priority:	High
Frequency of Use:	Infrequent (typically once per semester or academic year).
Normal Course of Events:	<ol style="list-style-type: none"> 1. The Department Secretary logs into the system. 2. The Secretary navigates to the "Curriculum Management" section. 3. The Secretary selects the "Import Curriculum" option. 4. The system prompts the Secretary to upload a curriculum file. 5. The Secretary uploads the file in the required format. 6. The system validates the file for correct structure and data consistency. 7. The system displays a confirmation message indicating a successful upload.
Alternative Courses:	<ul style="list-style-type: none"> • 3.7.AC.1: If the system detects errors in the uploaded file (e.g., missing or invalid data), it

	displays error messages and prompts the Secretary to correct the file and re-upload it.
Exceptions:	<ul style="list-style-type: none"> ● 3.7.EX.1 File Format Issues: If the uploaded file is in an unsupported format, the system rejects the file and prompts for re-upload. ● 3.7.EX.2 System Downtime: If the system is unavailable, the process cannot proceed.
Includes:	
Special Requirements:	<ul style="list-style-type: none"> ● The system must validate curriculum data against predefined rules to ensure accuracy. These rules include verifying that all course codes and names are unique, ensuring that total credit hours meet program requirements, checking the completeness of required fields such as course name, code, and credits, and validating that numeric fields like credit hours are positive integers. Additionally, the system must confirm that the file structure adheres to the specified format, including correct column headers and data organization. ● Support for standard curriculum file formats (e.g., Excel, CSV).
Assumptions:	The Department Secretary has the necessary technical knowledge to prepare and upload the file.
Notes and Issues:	<ul style="list-style-type: none"> ● Ensure file format specifications are clearly communicated to users. ● Consider implementing a preview feature to review curriculum data before finalizing the upload.

3.1.3.7.2 User Interface



Upload Required Documents
Choose a file or drag & drop it here

PDF, CSV, XLSX, and DOCX formats, up to 100MB

Browse File



underGraduateCurriculum.xls

25 MB of 35 MB • 🌟 Uploading...



Submit

- **Curriculum Management Interface:**
 - Upload button with file format guidelines.
 - Error message display for invalid file uploads.
 - Success confirmation message.
 - Option to view uploaded curriculum data in tabular format.

3.1.3.7.3 Performance/Software Quality Requirements

Response Times:

- File upload and validation: ≤ 5 seconds for files up to 1 MB.
- Curriculum preview: ≤ 3 seconds.

Data Accuracy:

- Course details from uploaded files must be 100% correctly mapped to the system.

System Availability:

- 99% availability during core hours (8AM-6PM).

Security:

- Only authorized users can upload files.

Scalability:

- Handle up to 5 concurrent uploads without performance degradation.

3.1.3.8 Submit Ranked List to Faculty Secretary**3.1.3.8.1 Use Case Template**

Use Case ID:	3.8		
Use Case Name:	Submit Ranked List to Faculty Secretary		
Created By:	Şahin Ökdem	Last Updated By:	Şahin Ökdem
Date Created:	18/11/24	Date Last Updated:	23/11/24

Actor:	Department Secretary
Description:	The Department Secretary submits a ranked list of students eligible for graduation, based on GPA, to the Faculty Secretary for approval and further processing. The Faculty Secretary may reject the ranked list with comments, prompting revisions and resubmission.

Preconditions:	<ul style="list-style-type: none"> • The Department Secretary must be logged into the system with appropriate permissions. • The ranked list must be prepared and verified for accuracy.
Postconditions:	<ul style="list-style-type: none"> • The ranked list is successfully submitted to the Faculty Secretary. • A record of the submission is stored in the system for future reference.
Priority:	High
Frequency of Use:	Infrequent (once per semester or academic year).
Normal Course of Events:	<ol style="list-style-type: none"> 1. The Department Secretary logs into the system. 2. The Secretary navigates to the "Ranked List Submission" section. 3. The Secretary selects the faculty to submit the list to. 4. The system prompts the Secretary to upload or generate the ranked list. 5. The Secretary reviews the list for accuracy. 6. The Secretary clicks "Submit to Faculty Secretary." 7. The system confirms successful submission and generates a submission receipt.
Alternative Courses:	<ul style="list-style-type: none"> • 3.8.AC.1: If the ranked list is not prepared, the system provides tools to generate the list from available student data. • 3.8.AC.2: If submission fails due to technical issues, the system suggests retrying or contacting support. • 3.8.AC.3: If the Faculty Secretary rejects the ranked list, they provide comments on required changes. The system notifies the Department Secretary, who revises and resubmits the list for approval.
Exceptions:	<ul style="list-style-type: none"> • 3.8.EX.1: File Format Issues: If the uploaded list does not meet system requirements, the system rejects it and prompts for corrections.


	<ul style="list-style-type: none"> ● 3.8.EX.2: System Downtime: If the system is unavailable, manual submission may be required.
Includes:	
Special Requirements:	<ul style="list-style-type: none"> ● The system should allow adding annotations or comments to the ranked list. ● Submissions should be time-stamped and trackable for audit purposes.
Assumptions:	The Faculty Secretary will review and approve the ranked list in a timely manner.
Notes and Issues:	<ul style="list-style-type: none"> ● Consider integrating automatic reminders for submission deadlines. ● Ensure compatibility with various file formats (e.g., Excel, PDF).

3.1.3.8.2 User Interface

logo

HomeRanked List SubmissionSend Docs

Profile




Upload Required Documents

Choose a file or drag & drop it here


PDF, CSV, XLSX, and DOCX formats, up to 100MB


Browse File


Generate New File



studentRankedList.xls

25 MB of 35 MB •  Uploading...



Select Faculty 

Submit

Ranked List Submission Interface:

- Option to upload or generate a ranked list.
- Preview of the ranked list with error-checking features.
- Submission button with a confirmation message.
- Submission history log with timestamps and receipt download.

3.1.3.8.3 Performance/Software Quality Requirements

Response Times:

- List upload or generation: ≤ 3 seconds.
- Submission processing: ≤ 2 seconds.

Data Accuracy:

- Submitted lists must match faculty secretary records with 100% accuracy.

System Availability:

- 99% availability during core hours (8AM-6PM).

Security:

- Only authorized users can manage and submit ranked lists.

Scalability:

- Handle up to 10 concurrent submissions without performance degradation.

3.1.3.9 Update student record**3.1.3.9.1 Use Case Template**

Use Case ID:	3.9		
Use Case Name:	Update student record		
Created By:	Şahin Ökdem	Last Updated By:	Şahin Ökdem
Date Created:	18/11/24	Date Last Updated:	23/11/24

Actor:	Department Secretary
Description:	The Department Secretary updates a student's academic or administrative record in the system, ensuring that all information is accurate and up to date for graduation eligibility.
Preconditions:	<ul style="list-style-type: none">• The Department Secretary must be logged into the system with appropriate permissions.• The student's record must exist in the system.

Postconditions:	<ul style="list-style-type: none"> • The student's record is successfully updated in the system. • A log of the changes made is stored for audit purposes.
Priority:	High
Frequency of Use:	Frequent (as needed for maintaining student records)..
Normal Course of Events:	<ol style="list-style-type: none"> 1. The Department Secretary logs into the system. 2. The Secretary navigates to the "Student Records" section. 3. The Secretary searches for the student by ID or name. 4. The system retrieves and displays the student's current record. 5. The Secretary updates the required fields (Completed courses, course letter grades, administrative holds). 6. The Secretary reviews the changes and clicks "Save." 7. The system validates the updates, saves the changes, and displays a confirmation message.
Alternative Courses:	<ul style="list-style-type: none"> • 3.9.AC.1: If the student's record is not found, the system provides options to retry the search or create a new record. • 3.9.AC.2: If validation fails (e.g., invalid data format), the system highlights the errors and prompts the Secretary to correct them.
Exceptions:	<ul style="list-style-type: none"> • 3.9.EX.1: If the Secretary lacks appropriate permissions, the system denies access and displays an error message. • 3.9.EX.2: If the system is unavailable, the record cannot be updated online. However, the Department Secretary may maintain a manual log of updates, which can be entered into the system once it becomes operational.

Includes:	
Special Requirements:	<ul style="list-style-type: none"> • The system must validate all updates to ensure data integrity. • A detailed change log must be maintained, including the time of update and the user who made the changes.
Assumptions:	<ul style="list-style-type: none"> • The Department Secretary has the correct and updated information to input into the system. • The system assumes that student records are unique and uses identifiers like student ID to prevent duplicate entries during creation or updates.
Notes and Issues:	<ul style="list-style-type: none"> • Ensure that sensitive information (e.g., personal identifiers) is protected during updates. • Consider implementing an approval process for certain types of updates (e.g., GPA adjustments).

3.1.3.9.2 User Interface

Student Records

View and update student records

Courses

Student Information

Search by student number or name

Filter

Records of student id 34123213823 returned

	COURSE NAME	Instructor	Completion Date	STATUS	Letter Grade	
<input type="checkbox"/>	Course1	Jane Cooper	2/19/21	Passed	AA	<div>Update</div>
<input type="checkbox"/>	Course2	Wade Warren	5/7/16	Passed	CB	<div>Update</div>
<input type="checkbox"/>	Course3	Esther Howard	9/18/16	Failed	FF	<div>Update</div>
<input type="checkbox"/>	Course4	Cameron Williamson	2/11/12	Passed	S	<div>Update</div>
<input type="checkbox"/>	Course5	Brooklyn Simmons	9/18/16	Failed	FD	<div>Update</div>
<input type="checkbox"/>	Course6	Leslie Alexander	1/28/17	Failed	W	<div>Update</div>
<input type="checkbox"/>	Course7	Jenny Wilson	5/27/15	Passed	BA	<div>Update</div>
<input type="checkbox"/>	Course8	Guy Hawkins	8/2/19	Passed	DD	<div>Update</div>

Save

Student Records Interface:

- Search functionality to locate a student by ID or name.
- Editable fields for all record details
- Confirmation dialog before saving updates.
- Log view showing historical changes to the student's record.

3.1.3.9.3 Performance/Software Quality Requirements

Response Times:

- Search and retrieve student record: ≤ 2 seconds.
- Save updated record: ≤ 2 seconds.

Data Accuracy:

- All updated fields must be validated with 100% accuracy.

System Availability:

- 99% availability during core hours (8AM-6PM).

Security:

- Only authorized users can update student records.

Scalability:

- Handle up to 20 concurrent updates without performance degradation.

3.1.4 Use Cases for Department Chair

3.1.4.1 Signing Cover Letter from Department Secretary

3.1.4.1.1 Use Case

Use Case ID:	4.1		
Use Case Name:	Signing Cover Letter from Department Secretary		
Created By:	Bekir Can Yalçın	Last Updated By:	Bekir Can Yalçın
Date Created:	17.11.2024	Date Last Updated:	22.11.2024

Actor:	Department Chair
Description:	The Department Chair reviews the cover letter from the Department Secretary to make sure it is accurate and complete, signs it, and sends it back to the Department Secretary.

Preconditions:	<ol style="list-style-type: none"> 1. Department Chair has logged into the system. 2. Department Secretary has uploaded the graduation cover letter and all graduation students documents into the system. 3. System has notified the department chair about the pending document for review. 4. Documents meet the initial requirements and have been verified by the department secretary.
Postconditions:	<ol style="list-style-type: none"> 1. Department Chair approves and digitally signs the cover letter through the system. 2. Signed Cover Letter is automatically sent back to the Department Secretary within the system. 3. If issues are identified, feedback is provided through the system, and the cover letter is returned to the department secretary for corrections.
Priority:	High
Frequency of Use:	Per Graduation Cycle


Normal Course of Events:	<ol style="list-style-type: none"> 1. Department Secretary uploads the graduation cover letter and related documents to the system. 2. The System sends a notification to the Department Chair, indicating a document is awaiting review. 3. Department Chair logs into the system and navigates to the "Pending Lists" section. 4. Department Chair opens the uploaded cover letter and supporting documents for review. 5. If the document is accurate and complete: 6. Department Chair approves and digitally signs the cover letter within the system. 7. The signed cover letter is automatically sent back to the Department Secretary.
Alternative Courses:	<p>4.1.AC. 1 – Not Signed Documents Transfer:</p> <ul style="list-style-type: none"> • System sends the document back to the Department Secretary without signed by Department Chair. <p>4.1.AC. 2 – Revised Documents Upload:</p> <ul style="list-style-type: none"> • The Department Secretary revises the document and re-uploads it to the system. <p>4.1.AC. 3 – Repetitive Process:</p> <ul style="list-style-type: none"> • The Department Chair repeats the review process.
Exceptions:	<p>4.1.EX. 1 – Technical Error:</p> <ul style="list-style-type: none"> • Department Chair is unable to access the system. <p>4.1.EX. 2 – Wrong Documents Uploaded:</p>

	<ul style="list-style-type: none"> • Department Chair controls and return the document to the department secretary for corrections.
Includes:	
Special Requirements:	<ol style="list-style-type: none"> 1. Secure login credentials for the graduation management system. 2. A user-friendly interface for reviewing and signing documents. 3. Notifications and tracking features to ensure timely task completion.
Assumptions:	<ol style="list-style-type: none"> 1. System is fully functional and accessible to all involved stakeholders. 2. Department Secretary has uploaded the correct and complete documents. 3. Department Chair has received training on using the system for these tasks.
Notes and Issues:	

3.1.4.1.2 User Interface

HomeLink1Link2Link3

Profile



Notifications: 1

General Information

Students List

Check Student Lis...

Additional Informati...

Home Screen

Q Search

Sort by ...

Possible Tasks

Sign Cover Letter and Check List

It will take a couple of minutes to change profile settings and confirm.

General Information

Students List

Cover Letters

Additional Informations

Q

Lesson Name	GPA	Contact	Details	Status
Student name 1	3.54	a@b.com	Show	Completed
Student name 2	2.87	a@b.com	Show	Completed
Student name 3	3.76	a@b.com	Show	Completed
Student name 4	2.15	a@b.com	Show	Completed
Student name 5	3.23	a@b.com	Show	Completed
Student name 6	1.87	a@b.com	Show	Not Completed

Cancel

Save Changes

Confirm List

Resend List

Sign Cover Letter and Check List

It will take a couple of minutes to change profile settings and confirm.

General Information

Students List

Cover Letters

Additional Informations

	Lesson Name	GPA	Contact	Details	Status
	Student name 1	3.54	a@b.com	Show	Completed
	Student name 2	2.87	a@b.com		Completed
	Student name 3	3.76	a@b.com		Completed
	Student name 4	2.15	a@b.com		Completed
	Student name 5	3.23	a@b.com		Completed
	<input type="text" value="Student name 6"/>	<input type="text" value="1.87"/>	<input type="text" value="a@b.com"/>		<input type="text" value="Not Completed"/>

Cover Letter 1

Student 1 in Computer Engineering want to Graduate with GPA 3.54; Completed Credits: 81. No missing courses, No Disciplinary Records, Eligible for Graduation. Tuition fee paid, 10/06/2026

290201032
Şahin Ökdem
sahinokdem@std.iyte.edu.tr

The Department Chair can see the Department Secretary's requests through notifications. From here, he can check the cover letter and student list sections, sign them and send them back.

3.1.5 Use Cases for Faculty Secretary

3.1.5.1 Inform Department Secretary

3.1.5.1.1 Use Case Template

Use Case ID:	5.1		
Use Case Name:	Inform Department Secretary		
Created By:	Bekir Can Yalçın	Last Updated By:	Bekir Can Yalçın
Date Created:	17.11.2024	Date Last Updated:	01.12.2024

Actor:	Faculty Secretary
Description:	Faculty Secretary reviews the ranked list of students eligible for graduation submitted by the Department Secretary. They verify its accuracy, provide feedback if necessary, and approve or reject the list for further processing.
Preconditions:	<ol style="list-style-type: none"> 1. The Faculty Secretary must be logged into the system with the appropriate permissions. 2. The ranked list must be submitted and accessible in the system.
Postconditions:	<ol style="list-style-type: none"> 1. Any feedback or rejection reasons are logged and communicated to the Department Secretary. 2. Ranked list is approved and forwarded to the dean for final processing.
Priority:	High
Frequency of Use:	Once per semester

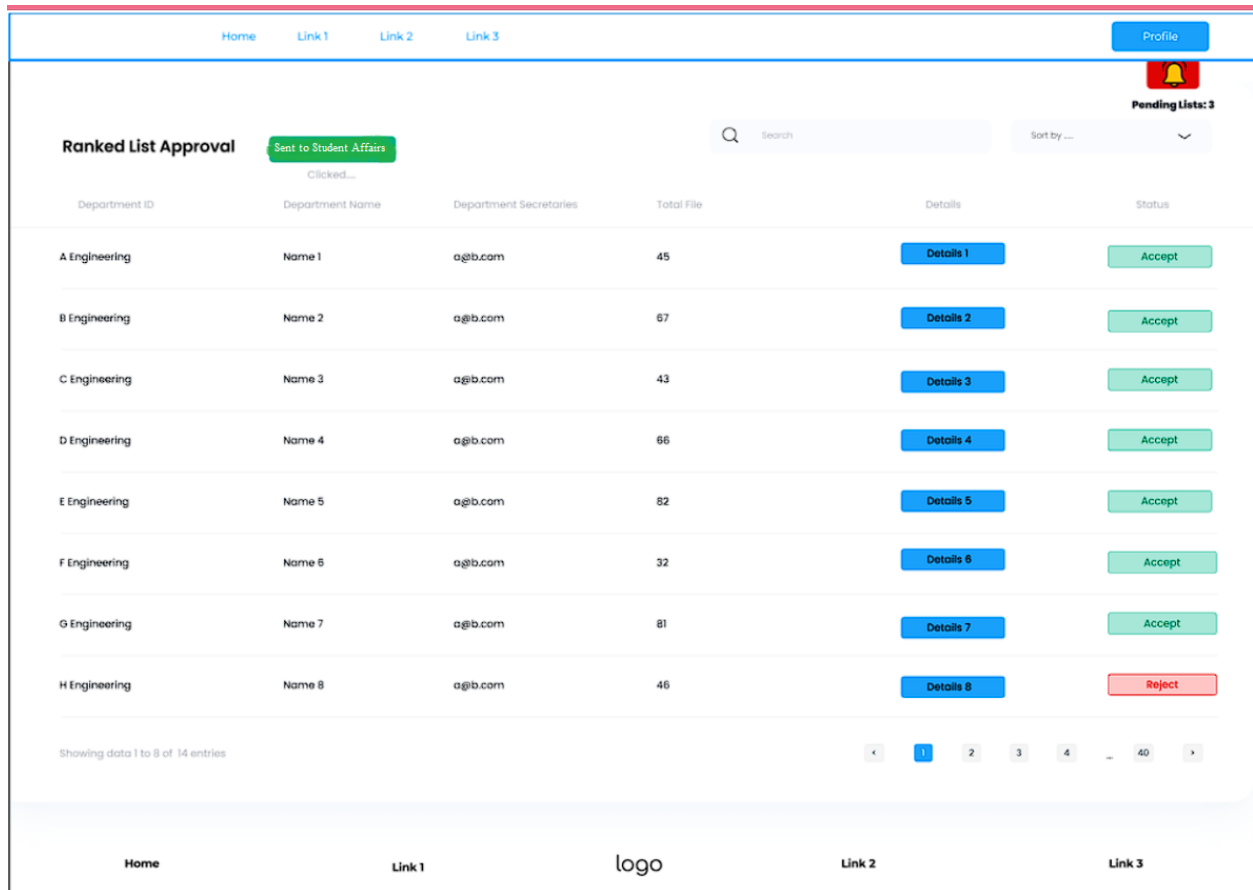
<p>Normal Course of Events:</p>	<ol style="list-style-type: none"> 1. The Faculty Secretary logs into the system. 2. The Secretary navigates to the "Ranked List Approval" section. 3. The system displays the ranked list submitted by the Department Secretary. 4. The Secretary reviews the list for accuracy and compliance with graduation policies. 5. If everything is in order, the Secretary clicks "Approve." 6. The system marks the list as approved and sends it to the dean. 7. The system generates a confirmation message and logs the approval.
<p>Alternative Courses:</p>	<p>5.1.AC.1 - If discrepancies are found in the ranked list:</p> <ul style="list-style-type: none"> • The Secretary clicks "Reject" and provides reasons for rejection. • The system logs the rejection and notifies the Department Secretary for corrections. <p>5.1.AC.2 - If additional information is required:</p> <ul style="list-style-type: none"> • The Faculty Secretary contacts the Department Secretaries via the UBYS.

Exceptions:	<p>5.1.EX.1 - Invalid List Format:</p> <ul style="list-style-type: none"> • The system rejects the list automatically and prompts the Secretary to contact the Department Secretary for re-submission. <p>5.1.EX.2 - System Downtime:</p> <ul style="list-style-type: none"> • The Secretary cannot review or approve the list and must defer the process.
Includes:	Use Case 3.3 for checking the list.
Special Requirements:	<ol style="list-style-type: none"> 1. The system must allow the Faculty Secretary to add comments during the review process. 2. The system should timestamp all actions for audit purposes.
Assumptions:	<ol style="list-style-type: none"> 1. The Faculty Secretary has the necessary knowledge of graduation requirements and policies. 2. The ranked list provided by the Department Secretary is prepared with the correct formatting and details.

Notes and Issues:

- Consider implementing a feature to automate reminders for pending approvals.
- Ensure the system complies with data privacy regulations for handling student records.

3.1.5.1.2 User Interface



When Faculty Secretary wants to see details for checking the Students list to determine Accept/Reject option.

Graduation List

Import New List

Export As



Search

Sort by : GPA



Student ID	Student Name	GPA	Email	Semester	Status
STU 001	Name 1	3.77	a@b.com	Semester 8	Ready
STU 002	Name 2	3.59	a@b.com	Semester 8	Ready
STU 003	Name 3	3.5	a@b.com	Semester 8	Ready
STU 004	Name 4	3.14	a@b.com	Semester 8	Ready
STU 005	Name 5	2.67	a@b.com	Semester 8	Not Ready
STU 006	Name 6	2.67	a@b.com	Semester 8	Ready
STU 007	Name 7	2.24	a@b.com	Semester 8	Ready
STU 008	Name 8	2.21	a@b.com	Semester 8	Ready

Showing data 1 to 8 of 3K entries

...

3.1.5.2 Send Graduation Decision to Faculties

3.1.5.2.1 Use Case Template

Use Case ID:	5.2		
Use Case Name:	Request Graduation Decision from DS.		
Created By:	Bekir Can Yalçın	Last Updated By:	Bekir Can Yalçın
Date Created:	17.11.2024	Date Last Updated:	02.12.2024

Actor:	Faculty Secretary
Description:	The Faculty Secretary sends the decision letter, signed by the dean, to the Student Affairs department. Student Affairs determines the graduation status of the students by reviewing the incoming documents and uploads the final decisions to the UBYS system, along with the ranked list after determining the student's rank.

Preconditions:	<ul style="list-style-type: none"> · The Faculty Secretary is logged into the system with appropriate permissions. · The ranked list of students has been reviewed and approved. · All necessary data (e.g. student details, graduation eligibility status) is complete and validated.
Postconditions:	<ul style="list-style-type: none"> · The faculties are notified of the graduation decision. · The graduation records are uploaded to the UBYS system by Student Affairs. · A log of the action is created for audit purposes.
Priority:	High
Frequency of Use:	Infrequent (once per semester).


<p>Normal Course of Events:</p>	<ol style="list-style-type: none"> 1. The Faculty Secretary logs into the system. <ul style="list-style-type: none"> ○ The system authenticates the user and displays the dashboard. 2. The Faculty Secretary navigates to the "Submit Decision Letter to Student Affairs" section. <ul style="list-style-type: none"> ○ A summary of section part are shown. 3. The Faculty Secretary uploads the decision letter signed by the dean to the system. <ul style="list-style-type: none"> ○ The system validates the uploaded document and displays a summary for review. 4. The Faculty Secretary reviews the details of the decision. <ul style="list-style-type: none"> ○ The Faculty Secretary confirms the upload. 5. The system sends the decision letter to Student Affairs and updates the status to "Sent to Student Affairs". <ul style="list-style-type: none"> ○ The system sends the decision letter to Student Affairs and updates the status to "Sent." 6. The system logs the action and displays a confirmation message.
<p>Alternative Courses:</p>	<p>5.2.AC. 1- Incomplete Data:</p> <ul style="list-style-type: none"> ● If some student data is incomplete, the system prevents the Faculty Secretary from sending the decision and provides an error message.

Exceptions:	<p>5.2.EX. 1 -System Downtime:</p> <ul style="list-style-type: none"> • The Faculty Secretary cannot upload the decision letter. The system notifies the user of the issue and prompts retry after resolution.
Includes:	<p>If necessary checks needed, Use Case 3.3 can help at that stage.</p>
Special Requirements:	<ul style="list-style-type: none"> • The system must timestamp the action for audit purposes. • Notifications should include a summary of the graduation decision for Student Affairs. • The panel must be free of unauthorized operations.
Assumptions:	<ol style="list-style-type: none"> 1. The Faculty Secretary has finalized the review and approval process. 2. Student Affairs is familiar with accessing graduation decisions via the system.
Notes and Issues:	<ul style="list-style-type: none"> • Consider adding an option to schedule sending decisions at a specific time. • Ensure data privacy compliance for handling student graduation records.



3.1.5.2.2 User Interface

[Home](#) [Link 1](#) [Link 2](#) [Link 3](#) [Profile](#)

**Pending Lists: 3**

Ranked List Approval

[Sent to Student Affairs](#)


Clicked...

Sort by ...

Department ID	Department Name	Department Secretaries	Total File	Details	Status
A Engineering	Name 1	a@b.com	45	Details 1	Accept
B Engineering	Name 2	a@b.com	67	Details 2	Accept
C Engineering	Name 3	a@b.com	43	Details 3	Accept
D Engineering	Name 4	a@b.com	66	Details 4	Accept
E Engineering	Name 5	a@b.com	82	Details 5	Accept
F Engineering	Name 6	a@b.com	32	Details 6	Accept
G Engineering	Name 7	a@b.com	81	Details 7	Accept
H Engineering	Name 8	a@b.com	46	Details 8	Reject

Showing data 1 to 8 of 14 entries

[1](#) [2](#) [3](#) [4](#) ... [40](#)

[Home](#) [Link 1](#)  [Link 2](#) [Link 3](#)

logo		Home	Graduation List	Profile	
Graduation List		Import New List	Export As	<div> <div>Q</div> <div>Search</div> </div>	Short by : GPA
Student ID	Student Name	GPA	Email	Semester	Status
STU 001	Name 1	3.77	a@b.com	Semester 8	Ready
STU 002	Name 2	3.59	a@b.com	Semester 8	Ready
STU 003	Name 3	3.5	a@b.com	Semester 8	Ready
STU 004	Name 4	3.14	a@b.com	Semester 8	Ready
STU 005	Name 5	2.67	a@b.com	Semester 8	Not Ready
STU 006	Name 6	2.67	a@b.com	Semester 8	Ready
STU 007	Name 7	2.24	a@b.com	Semester 8	Ready
STU 008	Name 8	2.21	a@b.com	Semester 8	Ready
Showing data 1 to 8 of 3K entries		<div> <div><</div> <div>1</div> <div>2</div> <div>3</div> <div>4</div> <div>...</div> <div>40</div> <div>></div> </div>			
Home		Link 1	logo	Link 2	Link 3

3.1.5.3 Combine graduating students from departments

3.1.5.3.1 Use case

Use Case ID:	5.3		
Use Case Name:	Manage Student Rankings and Consolidate Graduating Lists		
Created By:	Ege Ünlü	Last Updated By:	Ege Ünlü

Date Created:	19.11.2024	Date Last Updated:	24.11.2024
---------------	------------	--------------------	------------

Actor:	Faculty Secretary
Description:	The Faculty Secretary coordinates the consolidation of graduating student lists submitted by various departments. The system validates the completeness and accuracy of each department's submission and facilitates the integration of all departmental lists into a unified graduating student list for the faculty.
Preconditions:	<ul style="list-style-type: none"> • All departments have submitted their individual student rankings and graduating student lists. • Predefined ranking criteria, such as GPA and extracurricular activities, are configured in the system. • The system is operational and accessible by the Faculty Secretary.
Postconditions:	<ul style="list-style-type: none"> • The top three students are finalized and approved by the Faculty Secretary. • A unified, accurate graduating student list is consolidated and ready for publication.
Priority:	High
Frequency of Use:	Twice per academic year, during the graduation periods at the end of each semester.

Normal Course of Events:	<ol style="list-style-type: none"> 1. Departments submit their student rankings and graduating lists to the system. 2. The system validates the submissions for completeness and accuracy. 3. The system calculates the rankings based on predefined criteria. 4. The Faculty Secretary reviews the rankings and makes necessary adjustments. 5. The top three students are finalized and approved. 6. The validated graduating lists from all departments are merged into a unified faculty list. 7. The Faculty Secretary reviews and approves the consolidated graduating student list.
Alternative Courses:	<ul style="list-style-type: none"> ● Scenario 1: A department fails to submit its rankings or graduating list on time. <ul style="list-style-type: none"> ○ The system sends an automated reminder to the department. ● Scenario 2: A Faculty Secretary decides to include additional metrics in the ranking criteria. <ul style="list-style-type: none"> ○ The system updates the calculations to reflect the new criteria before finalizing rankings.
Exceptions:	<ul style="list-style-type: none"> ● Anticipated Errors: <ul style="list-style-type: none"> ○ <i>Error:</i> A department submits an incomplete or incorrect list. <ul style="list-style-type: none"> ■ System Response: The system flags the error and notifies the department for correction. ○ <i>Error:</i> Rankings include invalid or duplicate student entries. <ul style="list-style-type: none"> ■ System Response: The system prompts the Faculty Secretary to review and correct the entries. ● Unanticipated Failures: <ul style="list-style-type: none"> ○ <i>System Crash:</i> The system fails during processing.

	<ul style="list-style-type: none"> ■ System Response: Logs the failure, performs an automatic restart, and notifies IT support. ○ <i>Network Outage:</i> The system cannot access department submissions. <ul style="list-style-type: none"> ■ System Response: Saves progress locally and retries when the connection is restored.
Includes:	<p>Validate Departmental Submissions</p> <p>Calculate Rankings Based on Criteria</p>
Special Requirements:	<ul style="list-style-type: none"> ● The system must validate submissions within 24 hours to avoid delays. ● High performance is required to handle large data sets from multiple departments simultaneously.
Assumptions:	<ul style="list-style-type: none"> ● Departments are familiar with the submission process and adhere to deadlines. ● The ranking criteria are agreed upon and configured before the process begins.

3.1.5.3.2

User Interface

logo

[Home](#)[Graduation List](#)

Profile

Graduation List

Import New List

Export As

Q Search

Short by: GPA

Student ID	Student Name	GPA	Email	Semester	Status
STU 001	Name 1	3.77	a@b.com	Semester 8	Ready
STU 002	Name 2	3.59	a@b.com	Semester 8	Ready
STU 003	Name 3	3.5	a@b.com	Semester 8	Ready
STU 004	Name 4	3.14	a@b.com	Semester 8	Ready
STU 005	Name 5	2.67	a@b.com	Semester 8	Not Ready
STU 006	Name 6	2.67	a@b.com	Semester 8	Ready
STU 007	Name 7	2.24	a@b.com	Semester 8	Ready
STU 008	Name 8	2.21	a@b.com	Semester 8	Ready

Showing data 1 to 8 of 3K entries

<

1

2

3

4

...

40

>

Home

Link 1

logo

Link 2

Link 3

3.1.6.1 Rank Students Among the University

3.1.6.1.1 Use case

Use Case ID:	6.1		
Use Case Name:	Rank Students Among the University		
Created By:	Erdem Yetişen	Last Updated By:	
Date Created:	16/11/2024	Date Last Updated:	

Actor:	Student Affairs
Description:	The Student Affairs Department ranks students across the university according to their grade point averages and determines the best performing students and the top 3 in the university.
Preconditions:	<ol style="list-style-type: none"> 1. Student graduation status is confirmed. 2. The ranking period is active.
Postconditions:	<ol style="list-style-type: none"> 1. University-wide student ranking is generated and stored in the system. 2. The top 3 students are identified and notified. 3. A ranking report is prepared for download or further use.
Priority:	High
Frequency of Use:	Once per graduation period
Normal Course of Events:	<ol style="list-style-type: none"> 1. The Student Affairs officer logs into the system and accesses the "Student Ranking" module. 2. The officer selects the "Generate University Rankings" option. 3. The system retrieves academic records of all eligible students across departments. 4. The system calculates rankings based on GPA, resolving ties using secondary criteria such as the number of semesters in which students graduated. 5. The system identifies the top 3 students: <ul style="list-style-type: none"> • Applies tie-breaking criteria if there is a tie in the top ranks. • Marks the top 3 students as "Top Achievers." 6. The officer reviews the complete rankings and the top 3 student details in separate sections. 7. The officer confirms the rankings: <ul style="list-style-type: none"> • The system saves the finalized rankings. • Notifications are sent to the top 3 students and relevant departments. 8. The system generates a comprehensive ranking report, including a special section for the top 3 students.

	<p>9. The officer downloads the report for further use.</p> <p>10. The officer logs out, and the system secures the session.</p>
Alternative Courses:	<p>1.1.AC.1 - Manual Adjustment of Rankings:</p> <ul style="list-style-type: none"> • The officer identifies discrepancies or requires adjustments. • The officer edits the ranking order or overrides system-calculated results. • The system logs the manual changes and saves the adjusted ranking. <p>1.1.AC.2 - Tie for Third Place:</p> <ul style="list-style-type: none"> • If a tie occurs for the third position: <ul style="list-style-type: none"> ○ The system applies additional tie-breaking criteria, such as the student's total grade points or their performance in a specific subject or category relevant to the evaluation. ○ If a tie persists, the system flags the result for manual review by the officer. <p>1.1.AC.3 - Exclude Students:</p> <ul style="list-style-type: none"> • The officer excludes students from rankings based on administrative decisions. • The system adjusts the ranking order automatically.
Exceptions:	<p>1.1.EX.1 - Incomplete Data:</p> <ul style="list-style-type: none"> • If student records are incomplete or missing GPA values, the system displays an error message. • The officer contacts departments to update missing records before rerunning the ranking process. <p>1.1.EX.2 -System Error During Calculation:</p> <ul style="list-style-type: none"> • If the ranking process fails, the system logs the error and notifies technical support. • The officer can retry the operation after the issue is resolved.
Includes:	

Special Requirements:	<ol style="list-style-type: none"> 1. The ranking algorithm must support real-time calculations for large student datasets. 2. Reports must be exportable in multiple formats (PDF, Excel). 3. The system must highlight the top 3 students prominently in the ranking interface and reports. 4. Notifications to the top 3 students must include congratulatory messages and any relevant details about awards. 5. Ranking data must be secured to prevent unauthorized access or manipulation.
Assumptions:	<ol style="list-style-type: none"> 1. All student academic records are up to date in the system. 2. Student Affairs officers are trained to handle ranking operations.
Notes and Issues:	<ol style="list-style-type: none"> 1. Need to define tie-breaking criteria for ranking. 2. Clarify notification protocol for top-ranked students.

3.1.6.1.2 User Interface

logo

[Home](#)
[Link 1](#)
[Link 2](#)
[Link 3](#)

Profile

Graduation List

Import New List

Export As

Rank the students

Q

Search

Short by: GPA

▼

Student ID	Student Name	GPA	Email	Semester	Status
STU 001	Name 1	3.14	a@b.com	Semester 8	Ready
STU 002	Name 2	3.59	a@b.com	Semester 8	Ready
STU 003	Name 3	2.24	a@b.com	Semester 8	Ready
STU 004	Name 4	2.03	a@b.com	Semester 8	Ready
STU 005	Name 5	3.77	a@b.com	Semester 8	Ready
STU 006	Name 6	3.5	a@b.com	Semester 8	Ready
STU 007	Name 7	2.21	a@b.com	Semester 8	Ready
STU 008	Name 8	2.67	a@b.com	Semester 8	Not Ready

Showing data 1 to 8 of 3K entries

<

1

2

3

4

...

40

>

- Faculty secretary clicks the button “Rank the Students” next to the “Export As” button.
- The system will rank the students according to their GPA.

logo

Home

Graduation List

Profile

Graduation List

Import New List

Export As

Rank the students

Q

Search

Short by: GPA

Student ID	Student Name	GPA	Email	Semester	Status
<div></div> STU 001	Name 1	3.77	a@b.com	Semester 8	Ready
<div></div> STU 002	Name 2	3.59	a@b.com	Semester 8	Ready
<div></div> STU 003	Name 3	3.5	a@b.com	Semester 8	Ready
<div></div> STU 004	Name 4	3.14	a@b.com	Semester 8	Ready
<div></div> STU 005	Name 5	2.67	a@b.com	Semester 8	Not Ready
<div></div> STU 006	Name 6	2.67	a@b.com	Semester 8	Ready
<div></div> STU 007	Name 7	2.24	a@b.com	Semester 8	Ready
<div></div> STU 008	Name 8	2.21	a@b.com	Semester 8	Ready

Showing data 1 to 8 of 3K entries

<

1

2

3

4

...

40

>

Home

Link 1

logo

Link 2

Link 3

3.1.7.1.3 Performance/Software Quality Requirements

- Performance:
 - Page Load Time:** < 2 seconds for the dashboard and ranking results page.
 - Ranking Calculation:** Must complete calculations for 10,000+ students within 10 seconds.
 - Exporting Reports:** Generated ranking reports must be ready for download within 5 seconds.
 - Top 3 Selection:** Identify and highlight the top 3 students as part of the ranking process without adding more than 1 second to calculation time.
- Reliability:
 - Uptime:** 99.9% availability during peak graduation periods.
 - Data Backup:** All ranking data backed up daily.
 - Auto-Save:** Save progress every 2 minutes during manual ranking adjustments.
- Security:
 - Encryption:** Use SSL/TLS encryption for all data transmission.
 - Access Control:** Role-based access control to restrict ranking module access to authorized personnel.
 - Protection:** Prevent unauthorized editing or viewing of ranking data, especially for the top 3 students.
- Scalability:

- Handle ranking for up to 50,000 students during peak periods.

3.1.3.13 Use Case 4 for Student Affairs

3.1.3.13.1 Use case

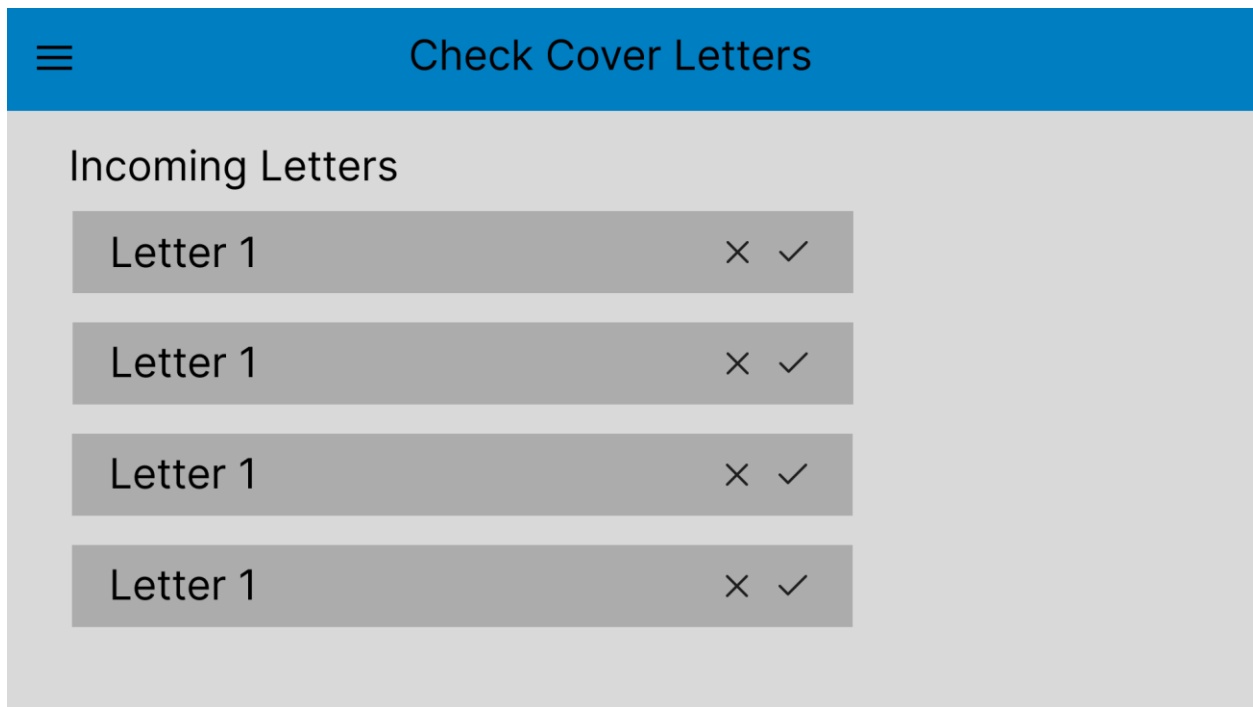
Use Case ID:	7.4		
Use Case Name:	Checking Faculty Cover Letter Submissions and Sending Warnings		
Created By:	Doğukan Doğan	Last Updated By:	
Date Created:	18.11.2024	Date Last Updated:	

Actor:	Student Affairs
Description:	This use case describes how student affairs staff checks whether all required cover letters for graduating students have been submitted by faculties. If any cover letters are missing, the staff sends warnings to the respective faculties to ensure timely submission.
Preconditions:	<ol style="list-style-type: none"> 1. The system contains a list of required cover letters tied to graduating students and faculties. 2. Faculties are registered in the system with up-to-date contact information. 3. Student affairs staff have access to the system for monitoring and sending notifications.
Postconditions:	<ol style="list-style-type: none"> 1. Missing cover letters are identified and tracked. 2. Faculties with pending submissions receive a warning notification. 3. The system logs all warnings sent for audit purposes.
Priority:	High

Frequency of Use:	During the pre-graduation preparation phase
Normal Course of Events:	<ol style="list-style-type: none"> 1. Student affairs staff logs into the system and navigates to the "Cover Letter Submission Status" section. 2. The system displays a list of faculties and their submission statuses. 3. The staff identifies faculties with missing cover letters. 4. The staff selects the "Send Warning" option for the respective faculties. 5. The system generates and sends a warning notification to the faculties via email or internal messaging. 6. The notification includes the content of the error and cause. 7. The system logs the warning action for tracking and accountability.
Alternative Courses:	<p>7.4 AC.1 Bulk Warnings:</p> <ul style="list-style-type: none"> • If multiple faculties have missing submissions, the staff can send warnings in bulk to save time. <p>7.4 AC.2 Manual Follow-Up:</p> <ul style="list-style-type: none"> • For repeated non-compliance, the staff may escalate the issue to faculties or initiate manual communication.
Exceptions:	<p>7.4 EX.1 Cover Letter Already Submitted:</p> <ul style="list-style-type: none"> • Cause: The system incorrectly marks a submission as missing. • System Response: Allow the staff to validate and mark the submission as complete. <p>7.4 EX.2 Faculty Contact Information Missing:</p> <ul style="list-style-type: none"> • Cause: Faculty contact details are incorrect or missing. • System Response: Notify the admin to update the contact details.
Includes:	<ol style="list-style-type: none"> 1. Monitoring module for cover letter submission status. 2. Notification system for sending warnings. 3. Escalation workflow for unresolved issues.

Special Requirements:	<ol style="list-style-type: none"> 1. Notifications must clearly state the missing cover letter details and the urgency of submission. 2. The system should provide a real-time status dashboard for easy monitoring. 3. Faculty members should be able to confirm receipt of the notification.
Assumptions:	<ol style="list-style-type: none"> 1. Faculties have been informed of the cover letter submission requirements and deadlines. 2. Student affairs staff are trained to use the monitoring tools in the system.
Notes and Issues:	<ol style="list-style-type: none"> 1. Consider implementing automated reminders for upcoming deadlines to minimize manual warnings. 2. A reporting tool for tracking submission rates and identifying frequent delays would be beneficial.

3.1.3.13.2 User Interface



3.1.3.13.3 Performance/Software Quality Requirements

1. System Performance

- **Response Time:** The system should be able to mark a cover letter as submitted or send a warning in under 3 seconds.
- **Load Time:** The faculty cover letter dashboard and faculty details should load in under 4 seconds, even with a large number of entries.

2. Scalability

- **Bulk Processing:** The system should support bulk updates, allowing staff to mark multiple cover letters as submitted or send warnings to multiple faculty members at once.

3.1.3.14 Use Case 5 for Student Affairs

3.1.3.14.1 Use case

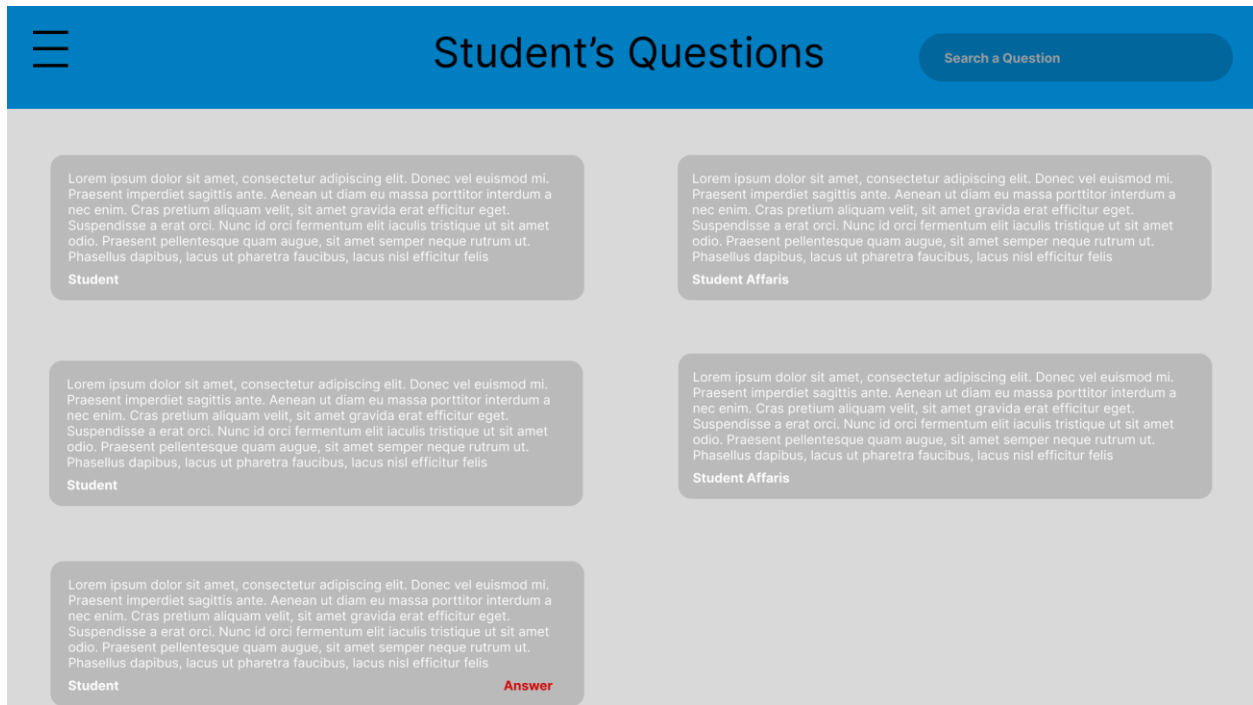
Use Case ID:	7.5		
Use Case Name:	Responding to Student Queries by Student Affairs.		
Created By:	Doğukan Doğan	Last Updated By:	
Date Created:	18.11.2024	Date Last Updated:	

Actor:	Student Affairs
Description:	This use case describes the process where the student affairs department receives, processes, and responds to students' questions regarding academic or administrative matters through the system.
Preconditions:	<ol style="list-style-type: none"> 1. The student has access to the system and is authenticated. 2. The student affairs staff has access to the system to view and respond to student queries. 3. A communication module exists within the system to handle student queries.
Postconditions:	<ol style="list-style-type: none"> 1. The student's question is acknowledged and answered. 2. A record of the query and response is stored in the system for reference. 3. The student receives a notification about the response.
Priority:	Medium.
Frequency of Use:	Daily.
Normal Course of Events:	<ol style="list-style-type: none"> 1. The student logs into the system and navigates to the "Ask a Question" section.

	<ol style="list-style-type: none"> The student submits a query, providing necessary details and selecting a category (e.g., enrollment, exam schedule, diploma process). The system logs the query and assigns it to the relevant student affairs staff member. The assigned staff member receives a notification and reviews the question. The staff member drafts and sends a response through the system. The student receives a notification with the response. The system logs the interaction for future reference.
Alternative Courses:	<p>7.5 AC.1 Self-Service Information:</p> <ul style="list-style-type: none"> If the question matches pre-defined FAQs, the system suggests answers automatically without involving staff. <p>7.5 AC.2 Escalation:</p> <ul style="list-style-type: none"> If the staff cannot resolve the query, it is escalated to higher authorities or specialized departments.
Exceptions:	<p>7.5 EX.1 Unclear Question Submission:</p> <ul style="list-style-type: none"> Cause: The student submits incomplete or vague details. System Response: Notify the student to provide additional details. <p>7.5 EX.2 Delayed Response:</p> <ul style="list-style-type: none"> Cause: High query volume or unavailable staff. System Response: Notify the student of delays and provide an estimated response time.
Includes:	<ol style="list-style-type: none"> Authentication and authorization mechanisms for students and staff. Notification system for updates on query progress. Query tracking and categorization for analytics and efficiency.
Special Requirements:	<ol style="list-style-type: none"> The system must be user-friendly and accessible to students with diverse needs.

	<ol style="list-style-type: none"> 2. Ensure privacy and confidentiality of student queries and responses.
Assumptions:	<ol style="list-style-type: none"> 1. Student affairs staff are trained to use the query management module. 2. Students have access to reliable internet and devices to use the system.
Notes and Issues:	<ol style="list-style-type: none"> 1. Introduce analytics to track query resolution times and improve staff efficiency. 2. Consider adding live chat support for urgent queries. 3. Regularly update FAQ content to reduce repetitive queries.

3.1.3.15.2 User Interface



3.1.3.15.3 Performance/Software Quality Requirements

1. System Performance

- **Response Time:** The query response should be submitted and acknowledged in under 3 seconds.
- **Load Time:** The query management dashboard and individual query details should load in under 4 seconds.

2. Scalability

- **Bulk Query Management:** The system should be capable of handling bulk queries, allowing staff to respond to or close multiple queries at once.

3.1.3.15 Use Case 6 for Student Affairs

3.1.3.15.1 Use case

Use Case ID:	7.6
--------------	-----

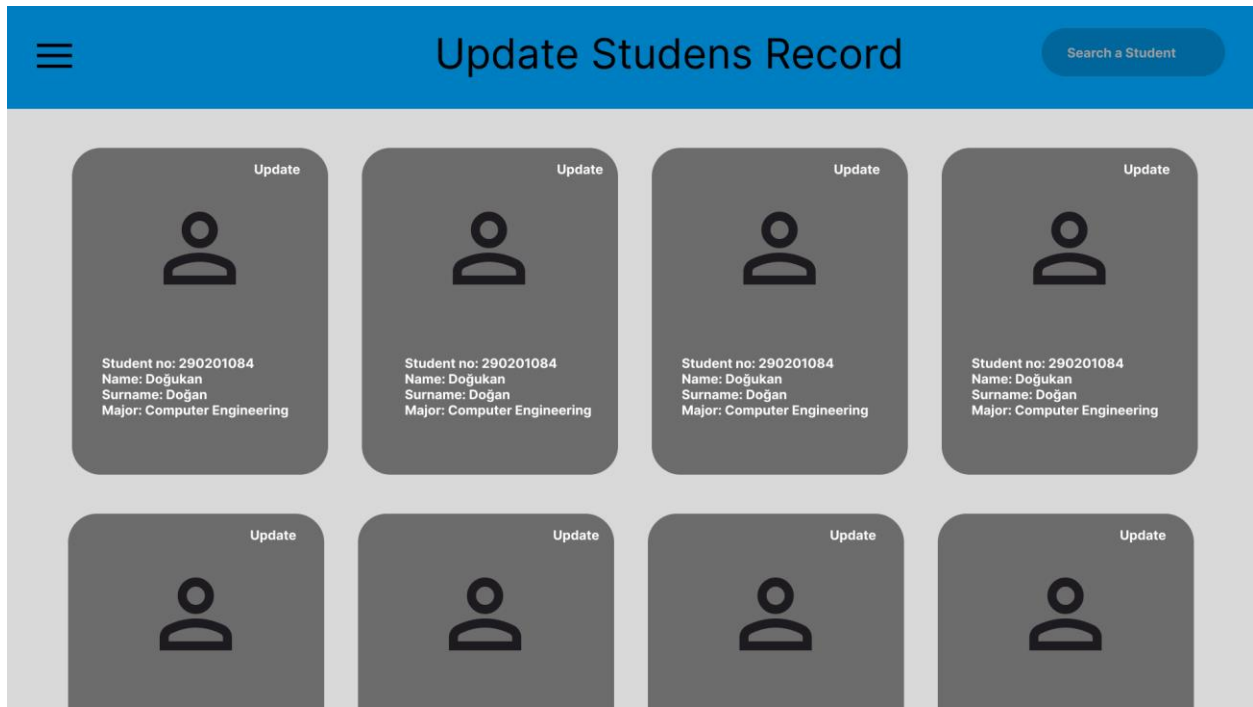
Use Case Name:	Updating Student Records by Student Affairs.		
Created By:	Doğukan Doğan	Last Updated By:	
Date Created:	18.11.2024	Date Last Updated:	

Actor:	Student Affairs
Description:	This use case describes the process where student affairs staff updates a student's record in the university's system to reflect changes such as contact information, enrollment status, academic standing, or other relevant details.
Preconditions:	<ol style="list-style-type: none"> 1. The student affairs staff is authenticated and authorized to access and modify student records. 2. The student record exists in the system.
Postconditions:	<ol style="list-style-type: none"> 1. The student record is successfully updated with accurate information. 2. A log entry is created, capturing the details of the update, including the staff member responsible. 3. The student is notified of the changes if applicable.
Priority:	High.
Frequency of Use:	Frequently, depending on student requests or administrative needs.
Normal Course of Events:	<ol style="list-style-type: none"> 1. The student affairs staff logs into the system using secure authentication. 2. The staff navigates to the "Student Records" section. 3. The staff searches for the specific student record by ID, name, or other identifiers. 4. The staff selects the record and initiates the update process.

	<ol style="list-style-type: none"> The system prompts the staff to edit the required fields (e.g., contact details, course enrollment, status). The staff verifies the changes and submits the update. The system validates the update for correctness and saves it. The system logs the changes and notifies relevant parties (e.g., the student or other departments).
Alternative Courses:	<p>7.7 AC.1 Validation Check Fails:</p> <ul style="list-style-type: none"> If the input data is invalid (e.g., incorrect date format), the system notifies the staff and highlights the errors for correction. <p>7.7 AC.2 Bulk Update:</p> <ul style="list-style-type: none"> If multiple records require similar updates, the system allows bulk changes using approved templates.
Exceptions:	<p>7.7 EX.1 Unauthorized Attempt:</p> <ul style="list-style-type: none"> Cause: The staff lacks proper permissions to update certain fields. System Response: Deny access and log the attempt for review. <p>7.7 EX.2 Record Not Found:</p> <ul style="list-style-type: none"> Cause: The student record does not exist or is archived. System Response: Notify the staff and suggest alternative actions, such as creating a new record or restoring archived data.
Includes:	<ol style="list-style-type: none"> Authentication and role-based access control. Audit trail for all changes made to student records.
Special Requirements:	<ol style="list-style-type: none"> The system must validate all updates to ensure data consistency (e.g., correct formats, non-duplicate entries). Secure data handling to protect student privacy and comply with regulations.

Assumptions:	<ol style="list-style-type: none"> 1. The staff has the required permissions and is trained to use the system. 2. The change request is legitimate and properly documented, if necessary.
Notes and Issues:	<ol style="list-style-type: none"> 1. Implement role-based restrictions to ensure sensitive fields (e.g., disciplinary records) are only editable by authorized personnel. 2. Consider an approval workflow for critical updates, such as changes to academic records or financial data. 3. Automate notifications to students for transparency in updates.

3.1.3.16.2 User Interface



3.1.3.16.3 Performance/Software Quality Requirements

1. System Performance

- **Response Time:** Record updates should save and reflect changes in less than 5 seconds.
- **Load Time:** The student record editing page should load in under 3 seconds, even with large data sets.

3.1.3.17 Use Case 1 for Rectorate

3.1.3.17.1 Use case

Use Case ID:	8.1		
Use Case Name:	Signing digital diplomas.		
Created By:	Doğukan Doğan	Last Updated By:	

Date Created:	18.11.2024	Date Last Updated:	
---------------	------------	--------------------	--

Actor:	Rectorate
Description:	The Rectorate signs all the diplomas online.
Preconditions:	<ol style="list-style-type: none"> 1. The graduation system is operational and accessible. 2. Diplomas for eligible students have been generated.
Postconditions:	<ol style="list-style-type: none"> 1. Diplomas are successfully signed with the rectorate's signature. 2. Signed diplomas are stored securely in the system and marked as ready for issuance.
Priority:	High
Frequency of Use:	Once per graduation period
Normal Course of Events:	<ol style="list-style-type: none"> 1. The rectorate logs into the graduation system using secure credentials. 2. The system displays the pending diplomas. 3. The rectorate selects diplomas for signing. 4. The system prints the diplomas for signing process 5. The rectorate reviews the document and applies their signature. 6. The system verifies the signature and updates the document status as "Signed." 7. A notification is sent to the relevant departments and students about the diploma's availability.
Alternative Courses:	<p>8.1 AC.1: If the rectorate identifies an issue with the diploma during the review:</p> <ul style="list-style-type: none"> • The document is flagged for revision and sent back to the issuing department.

	<ul style="list-style-type: none"> The e-signing process is postponed until corrections are made.
Exceptions:	<p>8.1 EX.1: The e-signature system is temporarily unavailable.</p> <ul style="list-style-type: none"> The rectorate is notified, and the process is rescheduled. The IT team is alerted to resolve the issue. <p>8.1 EX.2: The rectorate's e-signature certificate has expired.</p> <ul style="list-style-type: none"> The system alerts the rectorate to renew their certificate before proceeding. <p>8.1 EX.3: Diploma validation fails due to missing or incorrect information.</p> <ul style="list-style-type: none"> Cause: Discrepancy in student data or diploma format. System Response: Halt signing process and notify the responsible department.
Includes:	<ol style="list-style-type: none"> Authentication and authorization mechanisms for the rectorate. E-signature authentication mechanism.
Special Requirements:	<ol style="list-style-type: none"> The e-signature must comply with local legal and regulatory standards. High-security protocols to prevent unauthorized access or tampering.
Assumptions:	<ol style="list-style-type: none"> The rectorate is familiar with using the e-signature system. All necessary IT infrastructure (e.g., secure servers, e-signature software) is functional.
Notes and Issues:	<ol style="list-style-type: none"> Regular training should be provided to rectorate personnel on e-signature best practices. There should be contingency plans for manual signing in case of prolonged system downtime.

3.1.3.17.2 User Interface

Sign Diplomas	
Unsigned Diplomas	Signed Diplomas
Doğukan Doğan ✓	Bekir Can Yalçın ✗
Yiğit Kılınçer ✓	
Mert Ali Kızılarıslan ✓	
Şahin Ökdem ✓	
Furkan Satır ✓	
Erdem Yetişen ✓	
Ege Ünlü ✓	

3.1.3.17.3 Performance/Software Quality Requirements

1. System Performance

- **Response Time:** Signing a document should take less than 3 seconds.
- **Document Load Time:** Preview should load within 5 seconds.

3.1.3.18 Use Case 1 for User

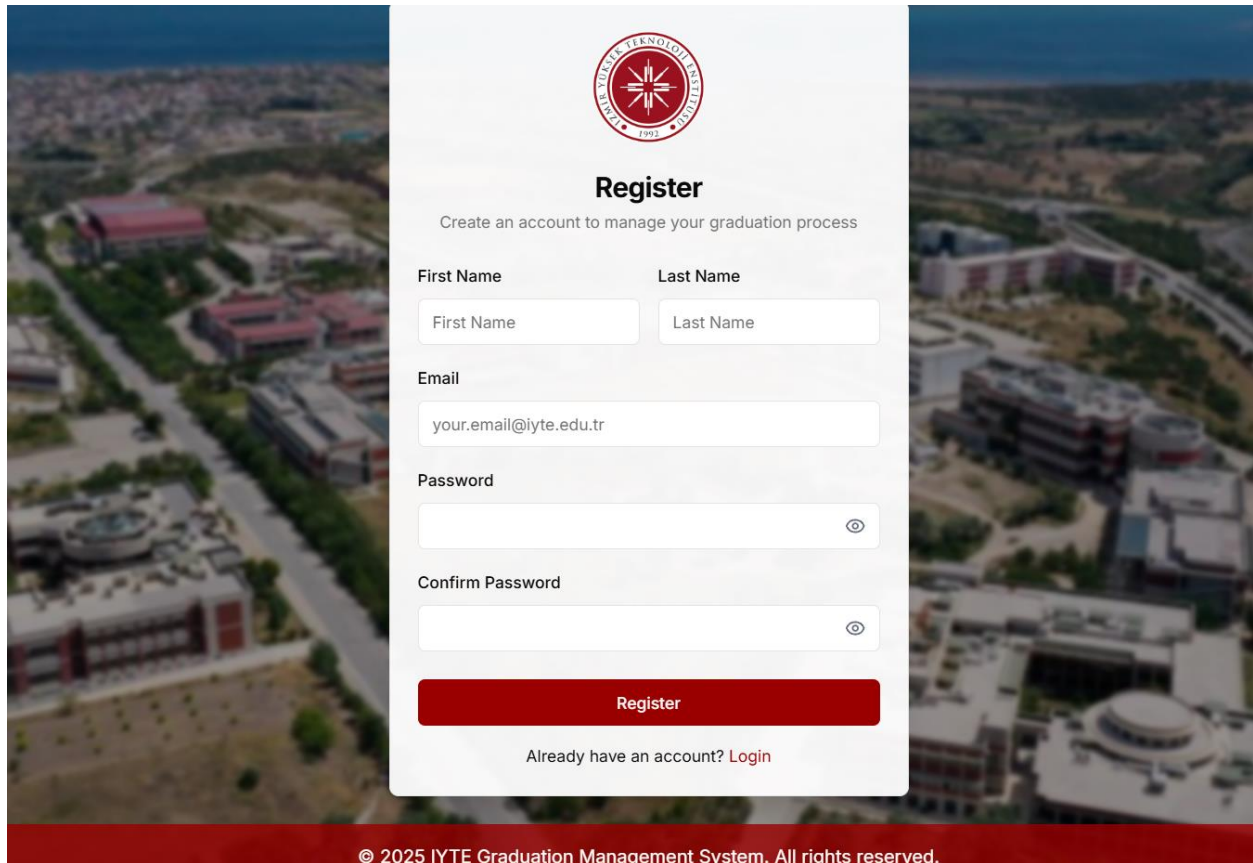
3.1.3.18.1 Use case

Use Case ID:	9.1		
Use Case Name:	Register		
Created By:	Furkan Satır	Last Updated By:	
Date Created:	18.11.2024	Date Last Updated:	

Actor:	User
Description:	A user accesses the registration page and registers for an account using a valid institutional email ending with @iyte.edu.tr. The system only accepts registrations from this domain, verifies the input, and upon successful registration, sends a confirmation email.
Preconditions:	<ul style="list-style-type: none"> User is not logged in. The registration page is accessible.
Postconditions:	<ul style="list-style-type: none"> A user account is created. A confirmation email is sent. The user is redirected to the confirmation/login page.
Priority:	High
Frequency of Use:	Frequent
Normal Course of Events:	<p>User opens the registration page: https://iytegms.vercel.app/register</p> <p>System loads the registration form.</p> <p>User fills in registration details with:</p> <ul style="list-style-type: none"> Email: <NEW_USER_@iyte.edu.tr> Password: Valid123 Confirm password: Valid123 <p>System verifies:</p> <ul style="list-style-type: none"> The email domain is @iyte.edu.tr. The form inputs are valid and passwords match. <p>If validation succeeds:</p> <ul style="list-style-type: none"> System registers the user. Sends a confirmation email.

	<ul style="list-style-type: none"> • Redirects to confirmation/login page with the message: "Confirmation link sent to your iyte.edu.tr email"
Alternative Courses:	<input type="checkbox"/> A1: Email domain is invalid <ul style="list-style-type: none"> • System shows an error message: "Only iyte.edu.tr emails are allowed for registration." <input type="checkbox"/> A2: Passwords do not match <ul style="list-style-type: none"> • System shows: "Passwords do not match." <input type="checkbox"/> A3: Missing fields or invalid format <ul style="list-style-type: none"> • System highlights errors and prevents submission.
Exceptions:	<ul style="list-style-type: none"> • E1: Network error while submitting <ul style="list-style-type: none"> ◦ System displays: "Registration failed. Please try again." • E2: Email server unavailable <ul style="list-style-type: none"> ◦ System logs error, retries email delivery, and shows: "We are experiencing delays sending the confirmation email."
Includes:	<input type="checkbox"/> Email validation logic <input type="checkbox"/> Confirmation email service
Special Requirements:	<input type="checkbox"/> Registration is restricted to emails ending with @iyte.edu.tr <input type="checkbox"/> Password strength rules must be enforced <input type="checkbox"/> Email confirmation must be sent within 1 minute
Assumptions:	<input type="checkbox"/> User has a valid @iyte.edu.tr email address <input type="checkbox"/> Mail server and registration service are operational
Notes and Issues:	

3.1.3.18.2 User Interface



Register
Create an account to manage your graduation process

First Name Last Name

Email

Password

Confirm Password

Register

Already have an account? [Login](#)

© 2025 IYTE Graduation Management System. All rights reserved.

3.1.3.18.3 Performance/Software Quality Requirements

1. System Performance

- **Response Time:** Signing a document should take less than 3 seconds.
- **Document Load Time:** Preview should load within 5 seconds.

3.1.3.19 Use Case 2 for User

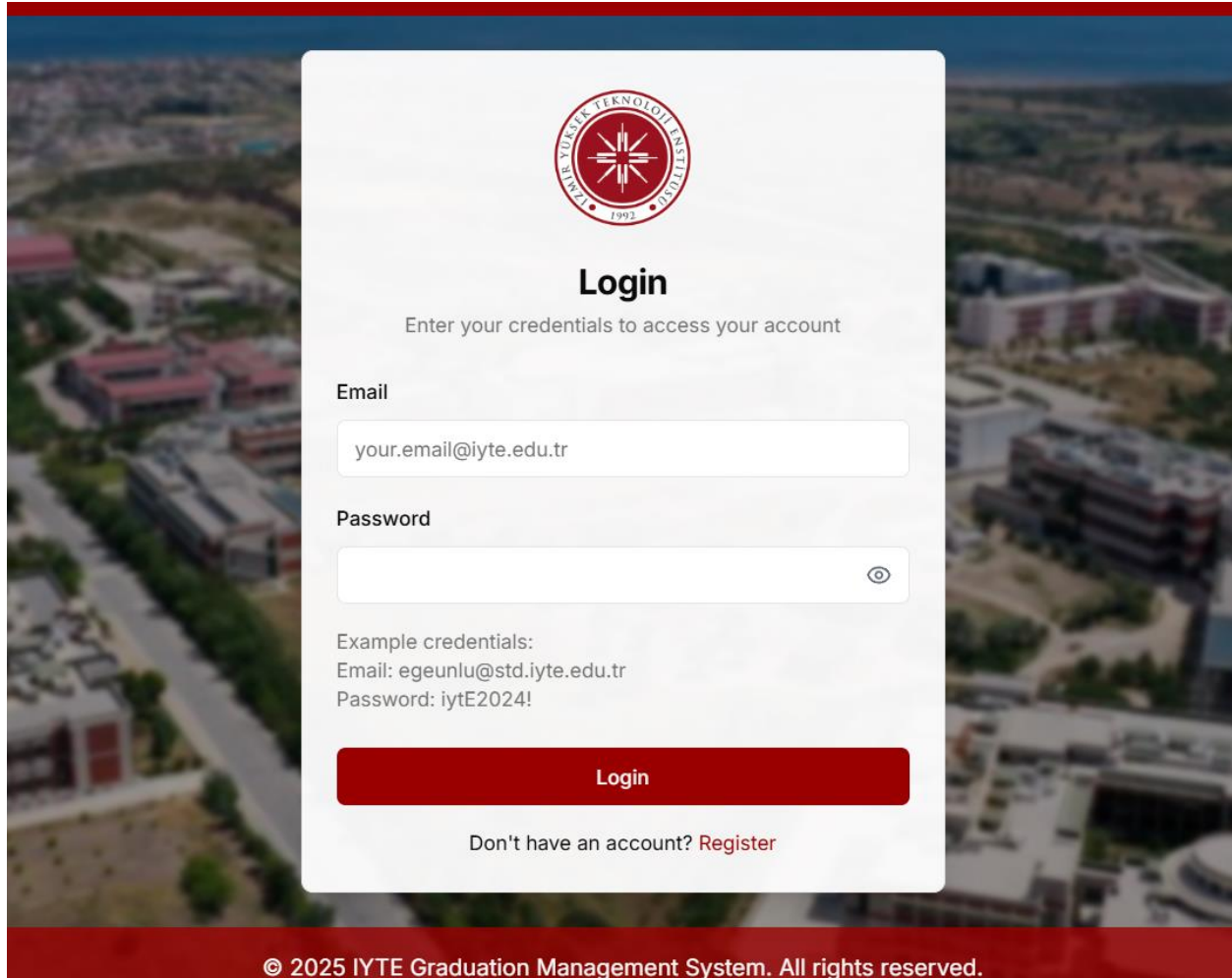
3.1.3.19.1 Use case


Use Case ID:	9.2		
Use Case Name:	Login		
Created By:	Furkan Satır	Last Updated By:	
Date Created:	18.11.2024	Date Last Updated:	

Actor:	User
Description:	A valid registered user opens the application and successfully logs in using their email and password. Upon successful authentication, the user is redirected to their dashboard.
Preconditions:	<ul style="list-style-type: none"> • A valid user account exists in the system • The login page is accessible
Postconditions:	<ul style="list-style-type: none"> • The user is authenticated • The user is redirected to the dashboard
Priority:	High
Frequency of Use:	Daily
Normal Course of Events:	<input type="checkbox"/> User opens the application: https://iytegms.vercel.app/login <input type="checkbox"/> System loads the login page <input type="checkbox"/> User enters valid credentials: <ul style="list-style-type: none"> • Email: <NEW_USER_EMAIL> • Password: Valid123 <input type="checkbox"/> User clicks the login button <input type="checkbox"/> System verifies credentials:

	<ul style="list-style-type: none"> • If valid → User is authenticated <input type="checkbox"/> User is redirected to the dashboard
Alternative Courses:	<input type="checkbox"/> A1: Invalid Email or Password <ul style="list-style-type: none"> • System displays: "Invalid login credentials." <input type="checkbox"/> A2: Account is not confirmed or active <ul style="list-style-type: none"> • System displays: "Please verify your email before logging in."
Exceptions:	<ul style="list-style-type: none"> • <input type="checkbox"/> E1: Network or Server Issue System shows: "Unable to connect. Please try again later." • <input type="checkbox"/> E2: Brute-force protection triggered System temporarily blocks login and shows: "Too many failed attempts. Please wait 15 minutes." <p style="text-align: center;">○</p>
Includes:	<input type="checkbox"/> Email/password input validation <input type="checkbox"/> User authentication and session handling
Special Requirements:	<input type="checkbox"/> Login must occur within 3 seconds under normal load <input type="checkbox"/> Password input must be securely masked
Assumptions:	<input type="checkbox"/> User remembers and correctly enters their credentials <input type="checkbox"/> Authentication service is online and operational
Notes and Issues:	

3.1.3.19.2 User Interface

The image shows a login interface for the IYTE Graduation Management System. At the top center is the IYTE logo, a red circular emblem with a white star-like symbol and the text 'İYTE YÜKSEK TEKNOLOJİ İSTİTÜTÜ' and '1992'. Below the logo is the heading 'Login' in bold black text, followed by the instruction 'Enter your credentials to access your account'. There are two input fields: 'Email' with the placeholder 'your.email@iyte.edu.tr' and 'Password' with a toggle icon. Below these fields, example credentials are provided: 'Email: egeunlu@std.iyte.edu.tr' and 'Password: iytE2024!'. A red 'Login' button is positioned below the examples. At the bottom, there is a link 'Don't have an account? Register'. The footer contains the copyright notice '© 2025 IYTE Graduation Management System. All rights reserved.'.



Login

Enter your credentials to access your account

Email

Password

Example credentials:
Email: egeunlu@std.iyte.edu.tr
Password: iytE2024!

Login

Don't have an account? [Register](#)

© 2025 IYTE Graduation Management System. All rights reserved.

3.1.3.19.3 Performance/Software Quality Requirements

1. System Performance

- **Response Time:** Signing a document should take less than 3 seconds.
- **Document Load Time:** Preview should load within 5 seconds.

3.1.3.20 Use Case 3 for User

3.1.3.20.1 Use case

Use Case ID:	9.3		
Use Case Name:	Info Page		
Created By:	Furkan Satır	Last Updated By:	
Date Created:	18.11.2024	Date Last Updated:	

Actor:	User
Description:	The system allows visitors to access the public information page without requiring authentication. The page is static and contains publicly viewable content.
Preconditions:	<input type="checkbox"/> Public Info Page is deployed and accessible <input type="checkbox"/> No login session or cookie is active
Postconditions:	<ul style="list-style-type: none">• The user is able to view the information page without logging in
Priority:	Occasionally(when users seek general information)
Frequency of Use:	Daily
Normal Course of Events:	<input type="checkbox"/> Visitor launches the application at: https://iytegms.vercel.app/ <input type="checkbox"/> System loads the login page or root route <input type="checkbox"/> Visitor ensures no login/session is active (no cookies or tokens present) <input type="checkbox"/> Visitor navigates to the Info Page

	<input type="checkbox"/> System allows access without authentication <input type="checkbox"/> Static info content is displayed
Alternative Courses:	A1: Info Page is moved or removed <ul style="list-style-type: none"> • System shows: "Page not found" or redirects to another public route
Exceptions:	<input type="checkbox"/> E1: Server not deployed or unreachable <ul style="list-style-type: none"> • System shows connection or 500 error <input type="checkbox"/> E2: Info Page is mistakenly restricted <ul style="list-style-type: none"> • System prompts for login (unexpected behavior)
Includes:	<input type="checkbox"/> Route configuration for anonymous access <input type="checkbox"/> Static content rendering
Special Requirements:	<input type="checkbox"/> No personal user data must be shown <input type="checkbox"/> Page must be fast-loading (under 2 seconds)
Assumptions:	<input type="checkbox"/> The Info Page is correctly deployed and not behind any authentication guard <input type="checkbox"/> Visitor has a browser with JavaScript enabled
Notes and Issues:	

3.1.3.20.2 User Interface

3.1.3.20.3 Performance/Software Quality Requirements

1. System Performance

- **Response Time:** Signing a document should take less than 3 seconds.
- **Document Load Time:** Preview should load within 5 seconds.

3.2 External Interfaces

The following describes all external interfaces, including inputs and outputs, for the system:

1. Email Notification System

- **Name of item:** Email Service
- **Purpose:** To send notifications to users regarding status updates, approvals, and reminders.
- **Source of input:** Graduation application system (e.g., student graduation status).
- **Destination of output:** Email inbox of students, advisors, and administrative staff.
- **Valid range/accuracy:** Email delivery must be completed within 2 minutes of the event triggering it.
- **Units of measure:** Time (minutes)
- **Timing:** Immediate delivery upon triggering events (e.g., application submission, status change).
- **Relationships to other inputs/outputs:** Dependent on system-triggered events for status updates.
- **Screen formats/organization:** Simple text-based notifications formatted according to user role.
- **Data formats:** Text, HTML for formatting.
- **Command formats:** SMTP protocol for email delivery.
- **End messages:** Confirmation of email sent successfully or error message if delivery fails.

3.3 Generic performance/software requirements

Static Requirements:

- **Number of Terminals to be Supported:**
 - **Normal Conditions:** The system must support up to 300 terminals simultaneously during regular business hours.
 - **Peak Conditions:** During critical graduation periods (e.g., graduation application submission deadlines), the system must support up to 1,000 terminals concurrently.
- **Number of Concurrent Users:**
 - **Normal Conditions:** The system must support 300 concurrent users performing typical tasks (e.g., viewing graduation status, updating records, submitting documents).

- **Peak Conditions:** During peak load periods (e.g., the last 2 weeks of the graduation application window), the system must support up to 1,000 concurrent users.

Dynamic Requirements:

- **Transaction Handling:**

- **Normal Conditions:** The system should handle 50 graduation application submissions per minute during routine periods (e.g., when students are not submitting applications in large numbers).
- **Peak Conditions:** During peak periods (e.g., the final month of the graduation application deadline), the system should be able to process 200 graduation application submissions per minute without degradation of performance.

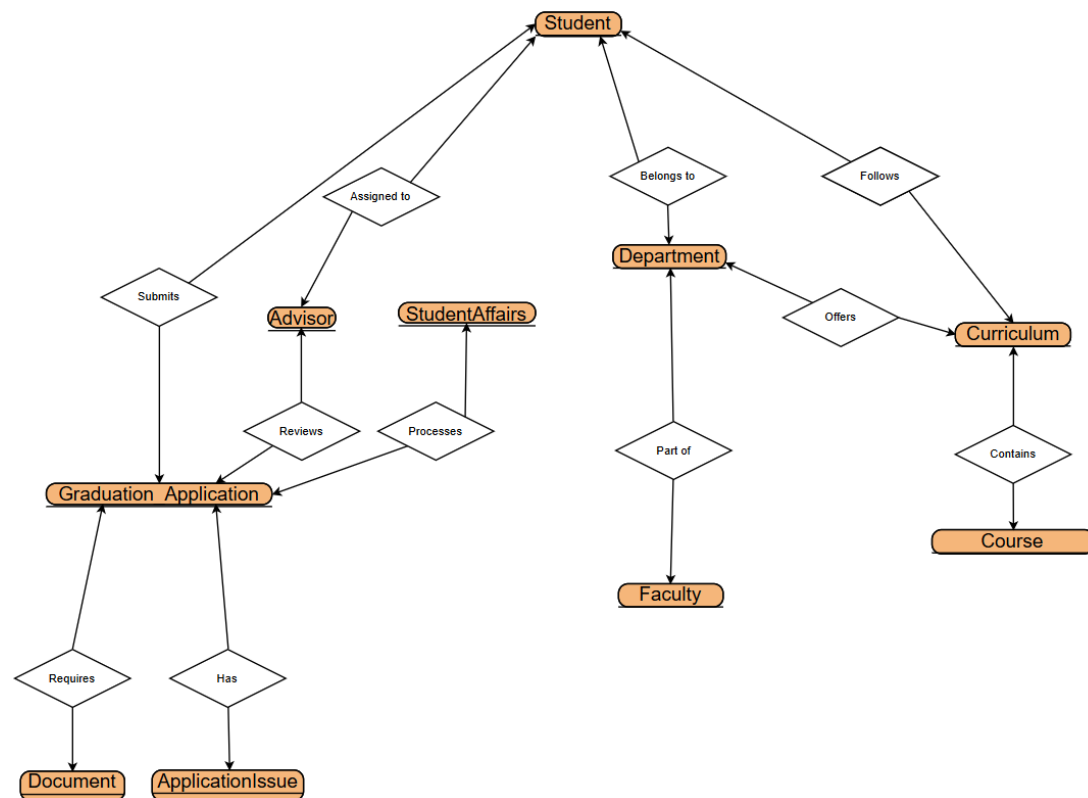
- **Response Time:**

- **Normal Conditions:** User interactions (e.g., checking graduation status, submitting applications) should have a response time of approximately 3 seconds or less.
- **Peak Conditions:** During periods of high activity, such as application submission windows, response time may increase to a maximum of approximately 5 seconds.

- **Data Processing Rate:**

- **Normal Conditions:** The system should be able to process up to approximately 100 graduation applications per minute (for validation and approval) during normal periods.
- **Peak Conditions:** During the peak graduation application season, the system must process up to approximately 1,000 applications per 5 minutes (including validation, approval, and document generation) without delays.

3.4 Logical Database Requirements



We only used connectors between entity sets so that the designer of the system is free to determine cardinalities.

3.5 Design Constraints

- Web-Based Application:**
 The system must be a web-based application to ensure accessibility from any device with internet connectivity. This design decision eliminates the need for installation on client machines, providing easier updates and maintenance.
- Responsive Design:**
 The system must support a responsive design to ensure it functions across different screen sizes, including desktops, tablets, and smartphones (e.g., major platforms, including Windows, macOS, iOS, and Android, across devices with screen resolutions from 480x800 pixels to 1920x1080 pixels). This is required to allow students, advisors, and staff to access the system from a variety of devices, both inside and outside the campus.
- Secure Authentication:**
 The system must implement secure user authentication using industry-standard methods (e.g., JWT, OAuth) to protect user data and ensure that only authorized users can access sensitive information. This design must comply with university data protection policies.

3.7 Other Requirements

- Audit logging system for mentioned actions in use cases using technologies like Kafka or RabbitMQ
- Document version control
- Data retention policies