Software Requirements Specification

for

Enhanced Course Feedback System

Version 1.0

Prepared by

Zekiye İpek Özmen 202028004
Emre Buğra Yaşar 202028021

Revision History

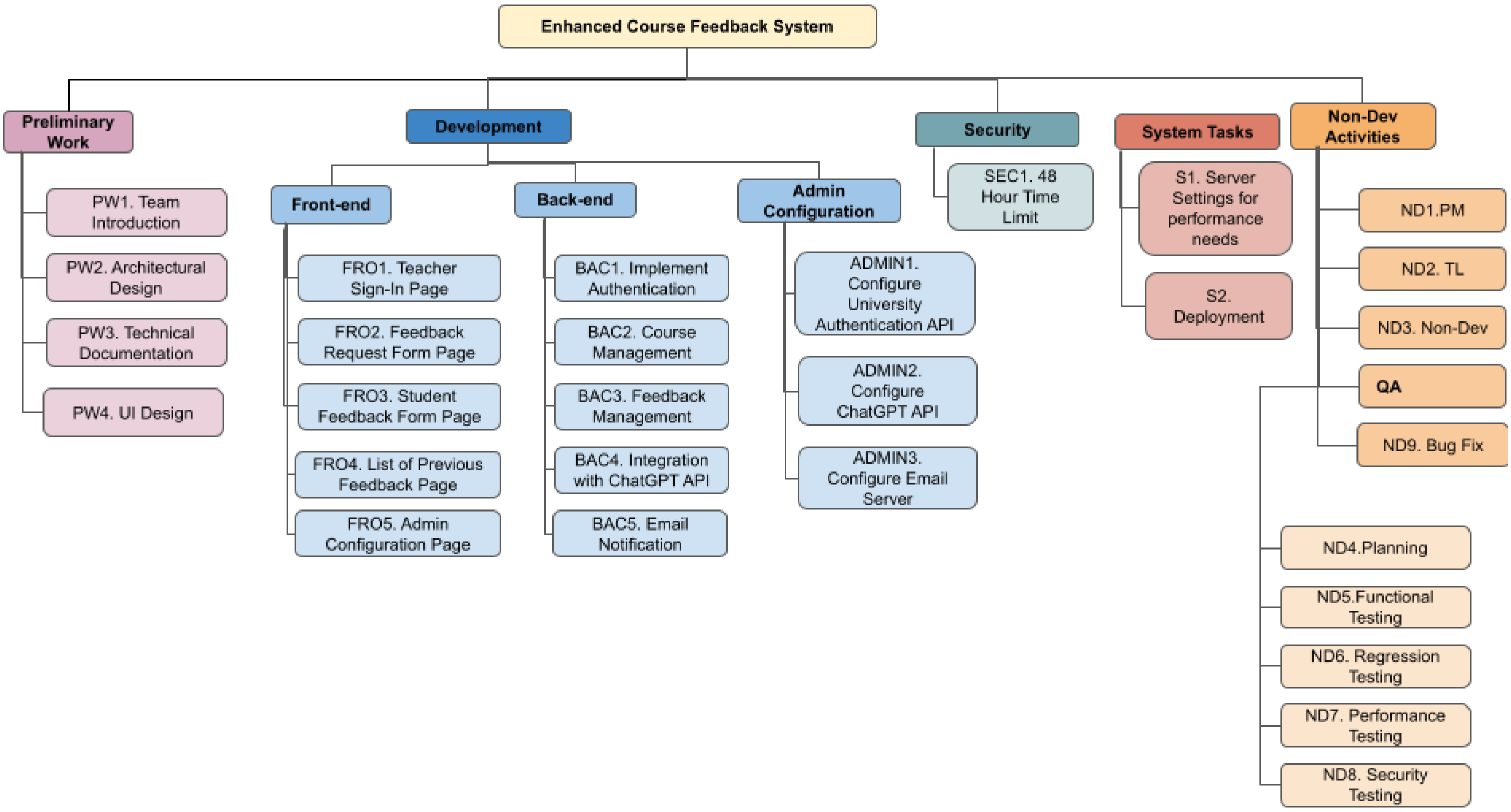
Name	Date	Reason For Changes	Version
SRS	16.12.2023	First Version Created	1.0

Table of Contents

1. RISK REGISTER TABLE

	Enhanced Course Feedback System Risk Register					
#	Risk Source	Probability of Risk	Impact on the project if risk source happens	Impact areas	Status	Mitigation Plan
1	Lack of Experience in Frontend Development	High	High	Project timeline, Frontend development progress	Not Occurred/In Progress	Provide training sessions for Satrio on frontend technologies.
2	Backend Development Complexity	Moderate	High	Project timeline, Backend development progress	Not Occurred/In Progress	Conduct a thorough analysis of backend requirements before development begins.
3	API Integration Issues	Moderate	Medium	Integration with university and ChatGPT APIs	Not Occurred/In Progress	Conduct early integration testing with university and ChatGPT APIs.
4	Quality Assurance Bottleneck	Moderate	Medium	QA process, Project timeline	Not Occurred/In Progress	Implement an iterative testing approach with continuous feedback to Anna.
5	E-mail Server Downtime	Low	Medium	Communication with teachers, Project timeline	Not Occurred/In Progress	Use a reliable e-mail service provider. Implement monitoring for server uptime.
6	University Authentication API Issues	Moderate	High	Authentication process, Project timeline	Not Occurred/In Progress	Engage with university IT to validate API integration early in the project.
7	Incomplete Feedback Summarization	Moderate	Medium	Feedback summarization accuracy, Project timeline	Not Occurred/In Progress	Conduct thorough testing of ChatGPT summarization before full implementation.

2. WBS


Enhanced Course Feedback System
Preliminary Work
PW1. Team Introduction
PW2. Architectural Design
PW3. Technical Documentation
PW4. UI Design
Development
Front-end
FRO1. Teacher Sign-In Page
FRO2. Feedback Request Form Page
FRO3. Student Feedback Form Page
FRO4. List of Previous Feedback Page
FRO5. Admin Configuration Page
Back-end
BAC1. Implement Authentication
BAC2. Course Management
BAC3. Feedback Management
BAC4. Integration with ChatGPT API
BAC5. Email Notification
Admin Configuration
ADMIN1. Configure University Authentication API
ADMIN2. Configure ChatGPT API
ADMIN3. Configure Email Server
Security
SEC1. 48 Hour Time Limit
System Tasks
S1. Server Settings for performance needs
S2. Deployment
Non-Dev Activities
ND1.PM
ND2. TL
ND3. Non-Dev
QA
ND9. Bug Fix
ND4.Planning
ND5.Functional Testing
ND6. Regression Testing
ND7. Performance Testing
ND8. Security Testing

3. ESTIMATIONS AND DEPENDENCIES

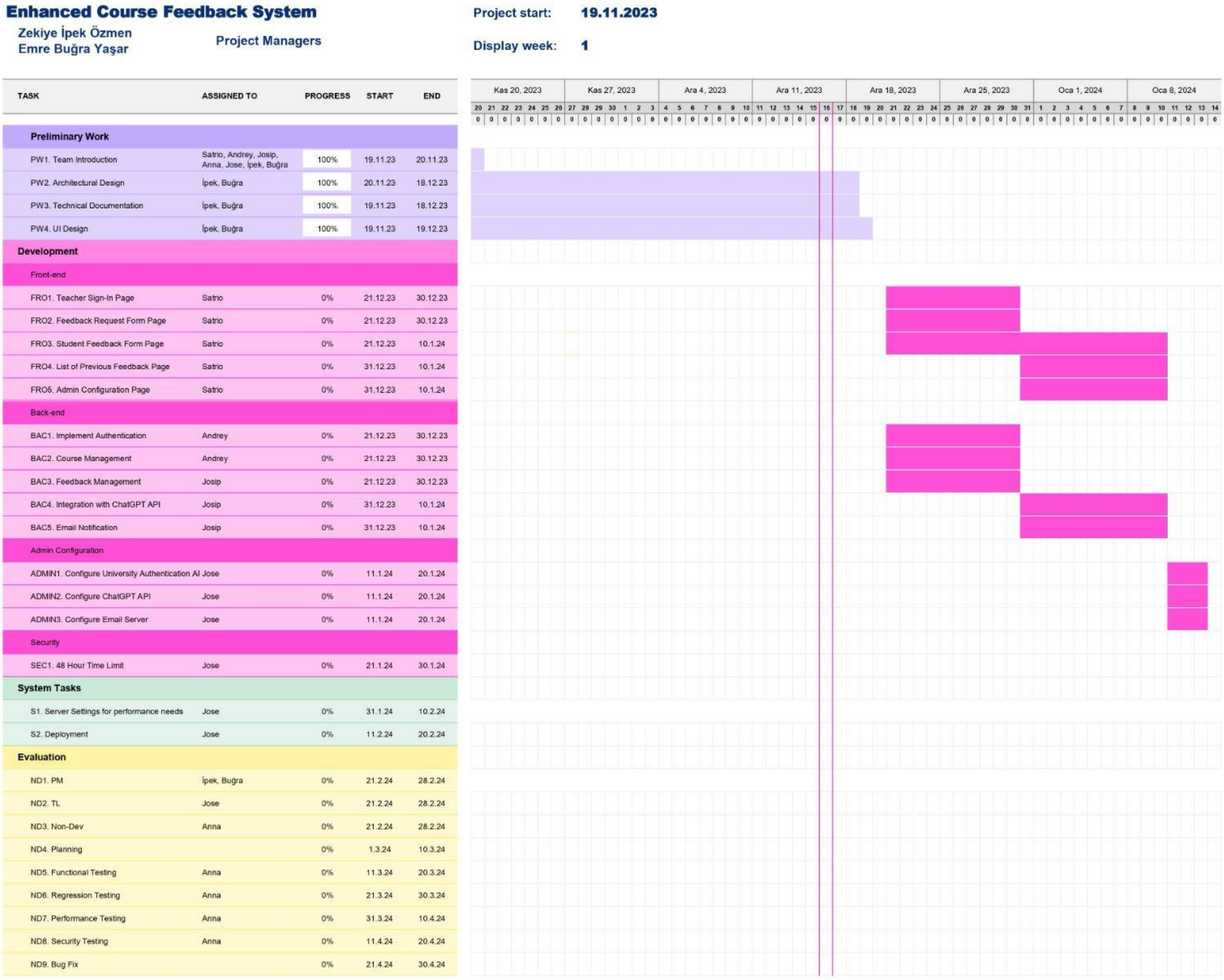
Enhanced Course Feedback System Dependencies		
Work Package	Task	Dependency
Preliminary Work	PW1. Team Introduction	
Preliminary Work	PW2. Architectural Design	PW1.
Preliminary Work	PW3. Technical Documentation	
Preliminary Work	PW4. UI Design	
Development/ FrontEnd Tasks	FRO1. Teacher Sign-In Page	PW3.
Development/ FrontEnd Tasks	FRO2. Feedback Request Form Page	PW3.
Development/ FrontEnd Tasks	FRO3. Student Feedback Form Page	PW3.
Development/ FrontEnd Tasks	FRO4. List of Previous Feedback Page	FRO1.
Development/ FrontEnd Tasks	FRO5. Admin Configuration Page	FRO2.
Development/ BackEnd Tasks	BAC1. Implement Authentication	
Development/ BackEnd Tasks	BAC2. Course Management	BAC1.
Development/ BackEnd Tasks	BAC3. Feedback Management	PW3.
Development/ BackEnd Tasks	BAC4. Integration with ChatGPT API	PW3.
Development/ BackEnd Tasks	BAC5. Email Notification	BAC3.
Development/ Admin Configuration Tasks	ADMIN1. Configure University Authentication API	
Development/ Admin Configuration Tasks	ADMIN2. Configure ChatGPT API	
Development/ Admin Configuration Tasks	ADMIN3. Configure Email Server	PW3.
Security	SEC1. 48 Hour Time Limit	
System Tasks	S1. Server Settings for performance needs	BAC3.
System Tasks	S2. Deployment	S1.
Non-Dev Tasks	ND1. PM	
Non-Dev Tasks	ND2. TL	ND1.
Non-Dev Tasks	ND3. Non-Dev	PW3.
QA	ND4. Planning	PW3.
QA	ND5. Functional Testing	ND4.
QA	ND6. Regression Testing	ND5.
QA	ND7. Performance Testing	S2.
QA	ND8. Security Testing	S2.
Non-Dev Tasks	ND9. Bug Fix	ND2.

(Dependencies table)

Enhanced Course Feedback System Estimates				
Task		Optimistic (hour)	Realistic (hour)	Pessimistic (hour)
Preliminary Work		39	50	70
PW1. Team Introduction		2	3	10
PW2. Architectural Design		20	25	30
PW3. Technical Documentation		5	7	10
PW4. UI Design		**12**	15	20
Development		**89**	114	162
Front-end				
FRO1. Teacher Sign-In Page		5	7	10
FRO2. Feedback Request Form Page		5	9	12
FRO3. Student Feedback Form Page		5	8	11
FRO4. List of Previous Feedback Page		6	7	10
FRO5. Admin Configuration Page		4	7	9
Back-end				
BAC1. Implement Authentication		7	8	13
BAC2. Course Management		6	7	10
BAC3. Feedback Management		8	9	11
BAC4. Integration with ChatGPT API		8	10	13
BAC5. Email Notification		6	9	12
Admin Configuration				
ADMIN1. Configure University Authentication API		8	10	16
ADMIN2. Configure ChatGPT API		10	11	15
ADMIN3. Configure Email Server		11	12	20
Security		6	10	15
SEC1. 48 Hour Time Limit		6	10	15
System Tasks		12	18	26
S1. Server Settings for performance needs		7	9	15
S2. Deployment		5	9	11
Total Work		146	192	273
ND1. PM	0.20	29.2	38.4	54.6
ND2. TL	0.20	29.2	38.4	54.6
ND3. Non-Dev	0.05	7.3	9.6	13.65
ND4. Planning	0.10	14.6	0.1	27.3
ND5. Functional Testing	0.20	29.2	38.4	54.6
ND6. Regression Testing	0.20	29.2	38.4	54.6
ND7. Performance Testing	0.20	29.2	38.4	54.6
ND8. Security Testing	0.20	29.2	38.4	54.6
ND9. Bug Fix	0.25	36.5	48	68.25
Grand Total		379.6	480.1	709.8

(Estimates table)

4. GANTT CHART



Enhanced Course Feedback System

Zekiye İpek Özmen
Emre Buğra Yaşar

Project Managers

Project start: **19.11.2023**

Display week: **1**

TASK	ASSIGNED TO	PROGRESS	START	END
Preliminary Work				
PW1. Team Introduction	Satrio, Andrey, Josip, Anna, Jose, İpek, Buğra	100%	19.11.23	20.11.23
PW2. Architectural Design	İpek, Buğra	100%	20.11.23	18.12.23
PW3. Technical Documentation	İpek, Buğra	100%	19.11.23	18.12.23
PW4. UI Design	İpek, Buğra	100%	19.11.23	19.12.23
Development				
Front-end				
FRO1. Teacher Sign-In Page	Satrio	0%	21.12.23	30.12.23
FRO2. Feedback Request Form Page	Satrio	0%	21.12.23	30.12.23
FRO3. Student Feedback Form Page	Satrio	0%	21.12.23	10.1.24
FRO4. List of Previous Feedback Page	Satrio	0%	31.12.23	10.1.24
FRO5. Admin Configuration Page	Satrio	0%	31.12.23	10.1.24
Back-end				
BAC1. Implement Authentication	Andrey	0%	21.12.23	30.12.23
BAC2. Course Management	Andrey	0%	21.12.23	30.12.23
BAC3. Feedback Management	Josip	0%	21.12.23	30.12.23
BAC4. Integration with ChatGPT API	Josip	0%	31.12.23	10.1.24
BAC5. Email Notification	Josip	0%	31.12.23	10.1.24
Admin Configuration				
ADMIN1. Configure University Authentication AI	Jose	0%	11.1.24	20.1.24
ADMIN2. Configure ChatGPT API	Jose	0%	11.1.24	20.1.24
ADMIN3. Configure Email Server	Jose	0%	11.1.24	20.1.24
Security				
SEC1. 48 Hour Time Limit	Jose	0%	21.1.24	30.1.24
System Tasks				
S1. Server Settings for performance needs	Jose	0%	31.1.24	10.2.24
S2. Deployment	Jose	0%	11.2.24	20.2.24
Evaluation				
ND1. PM	İpek, Buğra	0%	21.2.24	28.2.24
ND2. TL	Jose	0%	21.2.24	28.2.24
ND3. Non-Dev	Anna	0%	21.2.24	28.2.24
ND4. Planning		0%	1.3.24	10.3.24
ND5. Functional Testing	Anna	0%	11.3.24	20.3.24
ND6. Regression Testing	Anna	0%	21.3.24	30.3.24
ND7. Performance Testing	Anna	0%	31.3.24	10.4.24
ND8. Security Testing	Anna	0%	11.4.24	20.4.24
ND9. Bug Fix		0%	21.4.24	30.4.24

(To see the whole chart click to this link: https://1drv.ms/x/s!AnmEOlPl7lCJgolaInpe48gW5-iRWQ?e=SAuFPp)

5. INTRODUCTION

5.1. Purpose

The purpose of this document is to provide a detailed description of the Enhanced Course Feedback System, outlining its features, functionalities, and requirements. This document serves as a guide for developers and stakeholders involved in the project.

5.2. Scope of Project

The Enhanced Course Feedback System aims to streamline the feedback process between teachers and students at Çankaya University. The system will be a mobile app that allows teachers to request feedback, students to provide feedback, and integrates with ChatGPT for automated summarization.

5.3. Definitions, Acronyms, and Abbreviations

Term	Definition
Teacher	University staff responsible for course management.
Student	Individuals enrolled in a course.
ChatGPT	OpenAI's language model API for natural language processing.

5.4. References

[1] Çankaya University Official Website

5.5. Overview

This document includes sections on the overall description of the project, user characteristics, and a detailed specification of requirements. It serves as a comprehensive guide for the development team to understand the functionalities and features expected in the Enhanced Course Feedback System.

6. GENERAL DESCRIPTION

6.1. Product Perspective

The Enhanced Course Feedback System is a standalone mobile app designed to facilitate feedback communication between teachers and students. It integrates with external APIs, including the university's authentication API and ChatGPT for automated feedback summarization.

6.2. Product Functions

The system's main functions include:

- Teacher Feedback Request Form: Allows teachers to request feedback from students.
- Student Feedback Form: Allows students to provide feedback on courses.
- List of Previous Feedback: Provides a list of previous feedback.
- Admin Configuration: Allows the system administrator to configure API connections and system settings.

6.3. User Characteristic

6.3.1. Participants

Teachers: Faculty members using the system to request feedback.
Students: Enrolled individuals providing feedback on courses.

6.3.2. Admin

System Administrator: Responsible for configuring API connections and system settings.

6.4. General Constraints

Mobile Platform Dependency: The Enhanced Course Feedback System is designed as a mobile app, and its functionality is constrained by the capabilities and limitations of mobile devices.

Internet Connectivity: The system requires a stable internet connection to integrate with external APIs, such as the university's authentication API and ChatGPT. Lack of internet connectivity may impact the system's performance.

Device Compatibility: The system's UI/UX and overall performance are constrained by the diversity of mobile devices used by teachers and students. It must be compatible with a range of devices to ensure a consistent user experience.

Security and Privacy Regulations: The system must adhere to security and privacy regulations to safeguard user data. Compliance with university policies and data protection laws imposes constraints on data handling and storage.

Limited System Resources: The mobile devices used by teachers and students may have varying levels of resources (e.g., processing power, memory). The system must operate efficiently within these constraints to ensure optimal performance.

6.5. Assumptions and Dependencies

Assumption: University API Availability: It is assumed that the university's authentication API will be consistently available and accessible for integration throughout the development and operation of the system.

Assumption: ChatGPT API Accessibility: The system assumes continuous accessibility to the ChatGPT API for automated feedback summarization. Changes in the availability of the ChatGPT API may impact system functionality.

Dependency: External API Stability: The stability and reliability of external APIs, including the university's authentication API and ChatGPT API, are dependencies for the system's proper functioning. Any changes or disruptions to these APIs may require adjustments to the system.

Dependency: Internet Service Providers (ISPs): The system relies on the availability and reliability of internet services provided by ISPs. Any disruptions in internet services may impact the system's communication with external APIs.

Assumption: User Training: It is assumed that users, including teachers, students, and administrators, will receive adequate training on using the system. This assumption is crucial for the effective utilization of the feedback system.

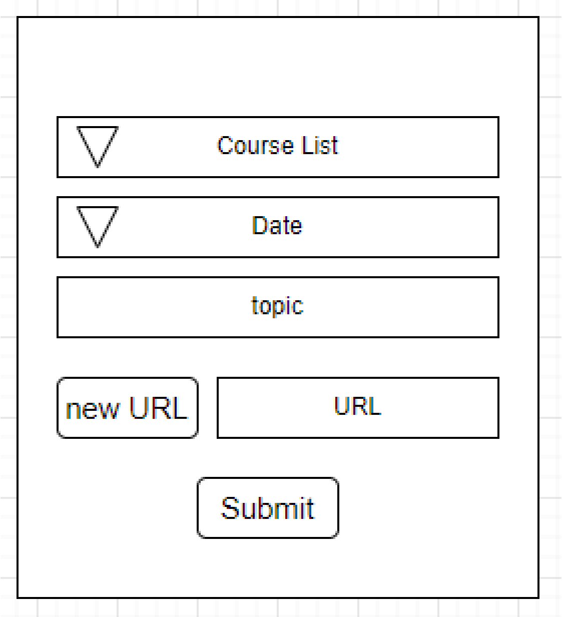
7. REQUIREMENTS SPECIFICATION

7.1. External Interface Requirements

7.1.1 UI Wireframes

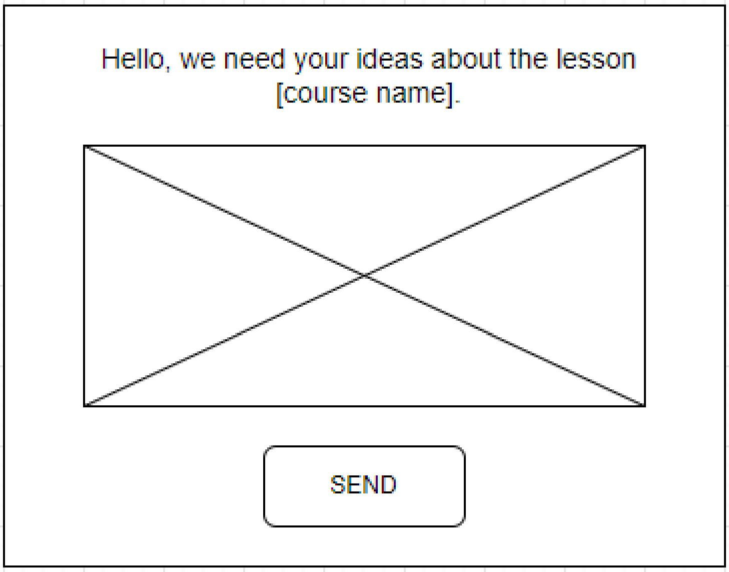
The system will have intuitive user interfaces for teachers, students, and administrators, allowing them to perform tasks seamlessly

7.1.1.1. Teacher Feedback Request Form



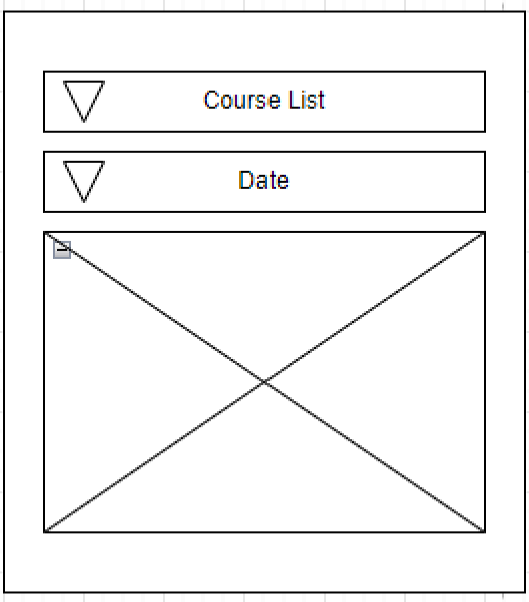
Teacher Feedback
Request From Interface

7.1.1.2. Student Feedback Form



Student Feedback Screen

7.1.1.3. List of previous feedback



Teacher See the Previous Feedbacks

7.1.2 Hardware interfaces

The system requires mobile devices for teachers and students. Additionally, integration with ChatGPT requires an internet connection.

7.1.3 Software interfaces

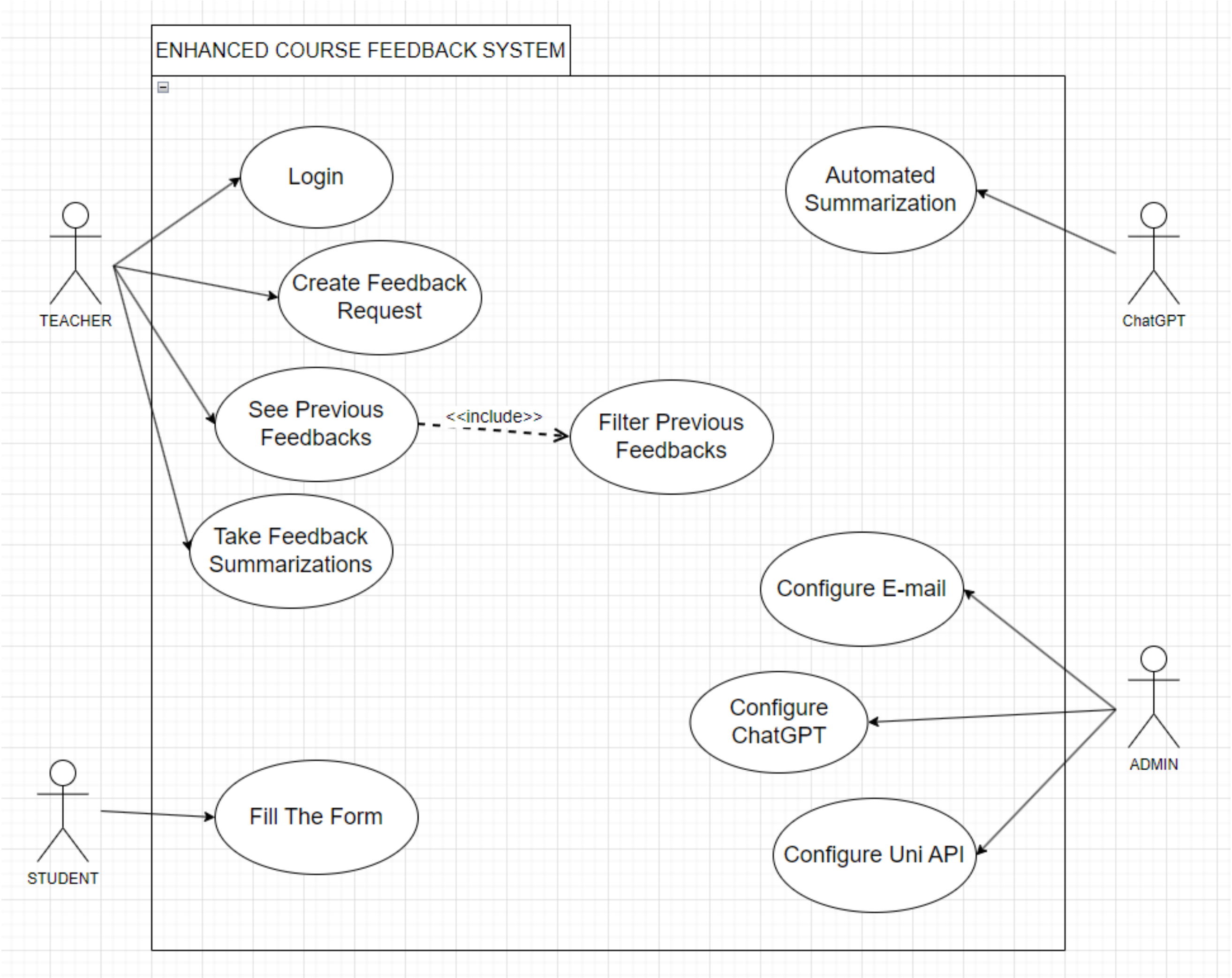
The system will interact with the university's authentication API, ChatGPT API, and an email server for communication purposes.

7.1.4 Communications interfaces

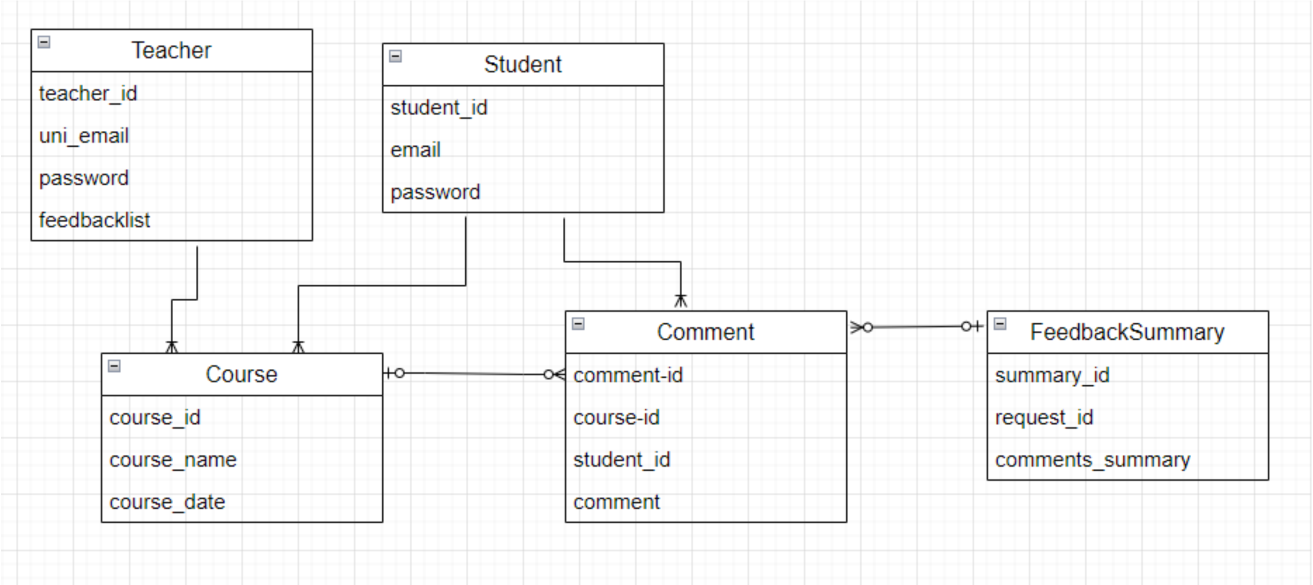
The system will utilize standard HTTP/HTTPS protocols for communication with external APIs.

7.2. Functional Requirements

7.2.1. Use-case diagram



7.2.2. Data Model



7.3. Stimulus/Response Sequences

7.3.1. Teacher Sign-in

Stimulus

1. Event: Teacher navigates to the sign-in page.
- Source: User (Teacher)
- Input: Accesses the sign-in page.

2. Event: Teacher enters university e-mail and password.
- Source: User (Teacher)
- Input: Provides university e-mail and password.

3. Event: Teacher clicks the "Sign In" button.
- Source: User (Teacher)
- Input: Initiates the sign-in process.

Response

1. Event: System verifies teacher credentials.
- Source: System
- Output: Checks the provided e-mail and password against the university's authentication API.

2. Event: System grants access to the teacher.

- Source: System
- Output: Allows access to the system if credentials are valid.

7.3.2. Teacher Feedback Request & Create URL

Stimulus

1. Event: Teacher selects a course from the list.
- Source: User (Teacher)
- Input: Chooses a course for feedback.

2. Event: Teacher picks a date from the calendar.
- Source: User (Teacher)
- Input: Selects a date for the feedback.

3. Event: Teacher enters topics for feedback.
- Source: User (Teacher)
- Input: Provides relevant topics.

4. Event: Teacher clicks the "Submit" button.
- Source: User (Teacher)
- Input: Initiates the feedback request and URL generation.

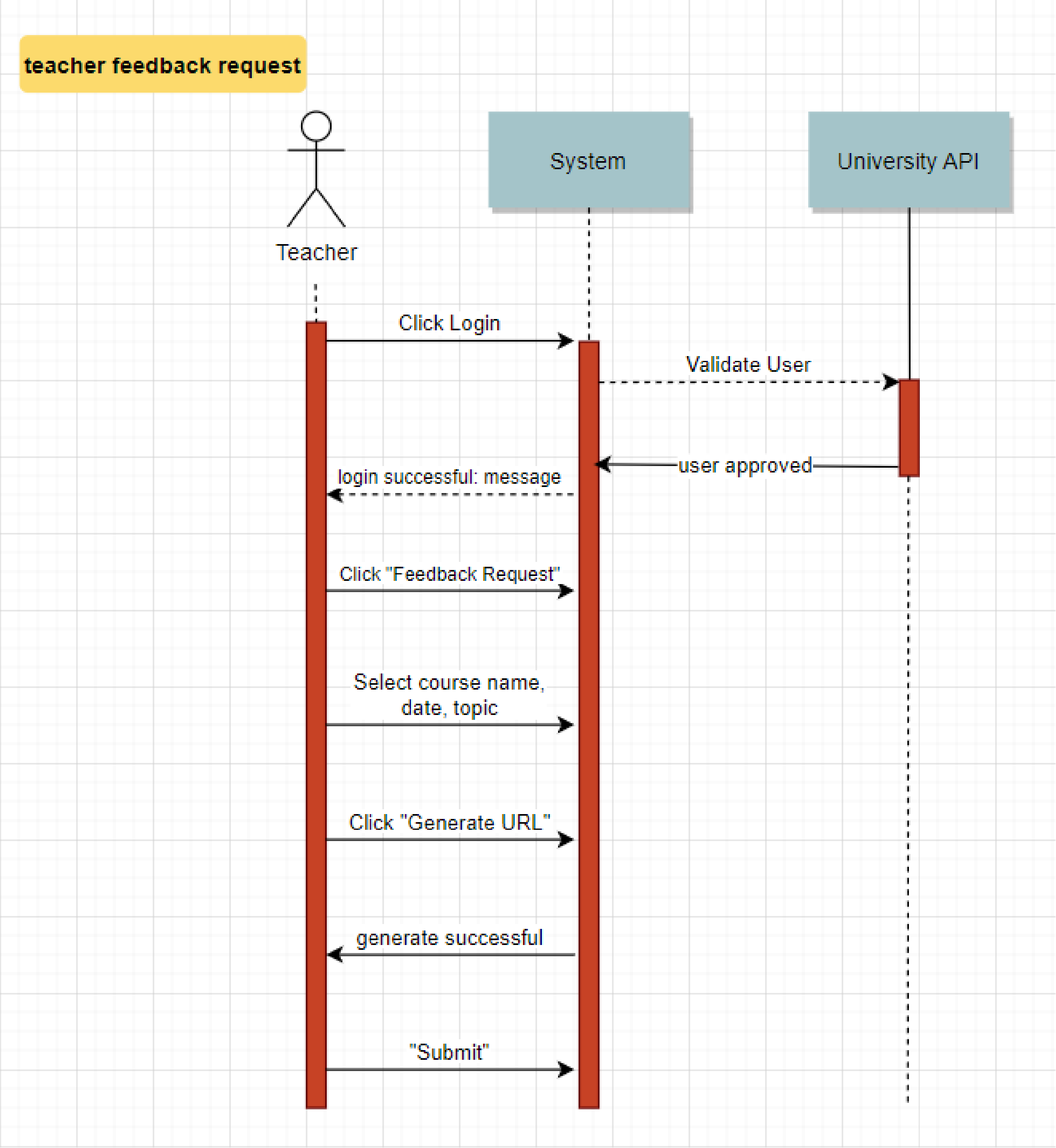
Response

1. Event: System validates selected course, date, and topics.
- Source: System
- Output: Checks for the correctness of entered information.

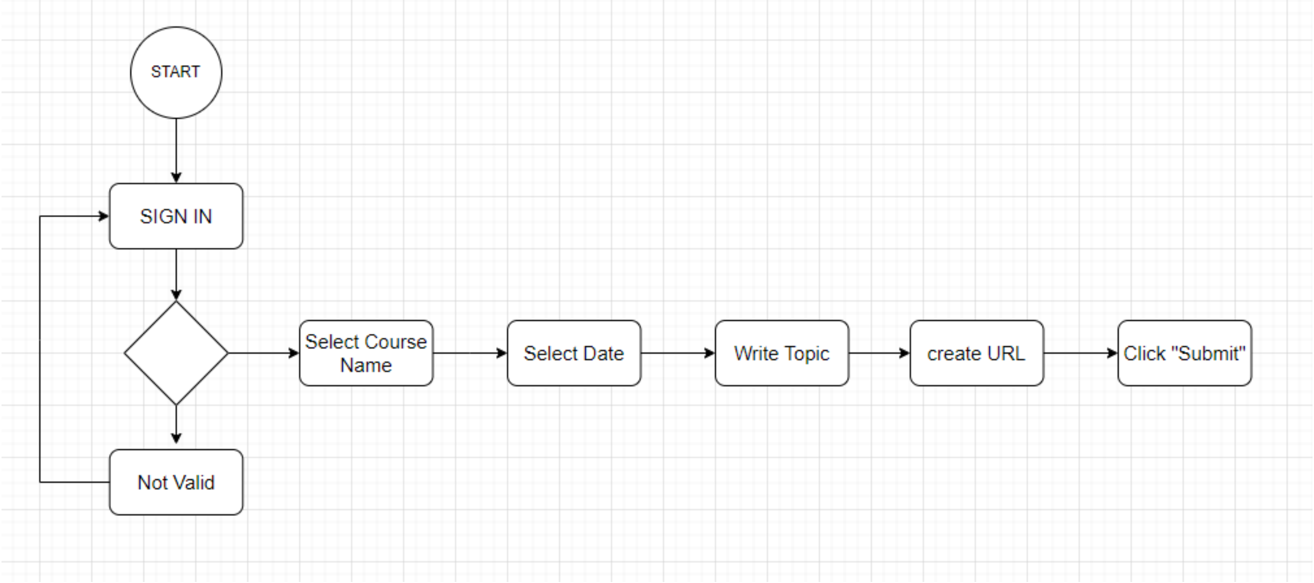
2. Event: System generates a unique URL.
- Source: System
- Output: Creates a unique URL associated with the provided feedback details.

3. Event: System sends the URL to registered students via e-mail.
- Source: System
- Output: Notifies students by sending the unique URL.

7.3.2.1. Sequence Diagram:Teacher Feedback Request & Create URL



7.3.2.2. Activity Diagram:Teacher Feedback Request & Create URL



7.3.3. Student : Inserting Feedback

Stimulus

1. Event: Student receives the unique URL via email.
- Source: System
- Input: Sends the unique URL to the student's registered email.

2. Event: Student clicks on the URL to access the feedback form.
- Source: User (Student)
- Input: Opens the URL provided in the email.

3. Event: Student provides feedback on course quality, content, and teaching methodology.
- Source: User (Student)
- Input: Submits feedback in the form.

4. Event: Student clicks the "Submit" button.
- Source: User (Student)
- Input: Initiates the submission of feedback.

Response

1. Event: System validates the unique URL.

- Source: System
- Output: Checks the authenticity of the URL.

2. Event: System accepts and records the feedback.
- Source: System
- Output: Stores the submitted feedback associated with the provided URL.

3. Event: System sets a 48-hour timer for URL expiration.
- Source: System
- Output: Initiates a timer for the provided URL to expire after 48 hours.

7.3.4. Feedback Delivery to the teacher

Stimulus

1. Event: System triggers ChatGPT API for feedback summarization.

- Source: System
- Input: Initiates the feedback summarization process.

2. Event: ChatGPT processes feedback and generates concise summaries.

- Source: ChatGPT API
- Input: Analyzes submitted feedback and generates summaries.

3. Event: System sends summarized feedback to the respective teacher via email.

- Source: System
- Output: Delivers summarized feedback to the teacher.

7.3.5. Admin will be able to configure the university authentication API connection info and credentials

Stimulus

1. Event: Admin navigates to the system configuration section.

- Source: User (Admin)
- Input: Accesses the configuration settings.

2. Event: Admin enters university authentication API connection information and credentials.

- Source: User (Admin)
- Input: Provides API connection details and credentials.

3. Event: Admin clicks the "Save Changes" button.

- Source: User (Admin)
- Input: Initiates the saving of configuration changes.

Response

1. Event: System validates and saves the provided information.

- Source: System
- Output: Verifies the correctness of API connection details and credentials and saves the configuration.

7.3.6. Admin will be able to configure the ChatGPT API connection info and credentials

Stimulus

1. Event: Admin navigates to the system configuration section.

- Source: User (Admin)
- Input: Accesses the configuration settings.

2. Event: Admin enters ChatGPT API connection information and credentials.

- Source: User (Admin)
- Input: Provides API connection details and credentials.

3. Event: Admin clicks the "Save Changes" button.

- Source: User (Admin)
- Input: Initiates the saving of configuration changes.

Response

1. Event: System validates and saves the provided information.

- Source: System
- Output: Verifies the correctness of ChatGPT API connection details and credentials and saves the configuration.

7.3.7. Admin will be able to configure the e-mail server connection info and credentials

Stimulus

1. Event: Admin navigates to the system configuration section.

- Source: User (Admin)
- Input: Accesses the configuration settings.

2. Event: Admin enters email server connection information and credentials.

- Source: User (Admin)
- Input: Provides email server connection details and credentials.

3. Event: Admin clicks the "Save Changes" button.

- Source: User (Admin)
- Input: Initiates the saving of configuration changes.

Response

1. Event: System validates and saves the provided information.
- Source: System
- Output: Verifies the correctness of email server connection details and credentials and saves the configuration.

7.3.8. Admin Configuration

Stimulus

1. Event: Admin navigates to the system configuration section.
- Source: User (Admin)
- Input: Accesses the configuration settings.

2. Event: Admin enters configuration details for university authentication API.
- Source: User (Admin)
- Input: Provides API connection details and credentials.

3. Event: Admin enters configuration details for ChatGPT API.
- Source: User (Admin)
- Input: Provides API connection details and credentials.

4. Event: Admin enters configuration details for the email server.
- Source: User (Admin)
- Input: Provides email server connection details and credentials.

5. Event: Admin clicks the "Save Changes" button.
- Source: User (Admin)
- Input: Initiates the saving of configuration changes.

Response

1. Event: System validates and saves the provided information.
- Source: System
- Output: Verifies the correctness of API connections and credentials and saves the configuration.

7.3.9. Listing of previous feedback

Stimulus

1. Event: Teacher navigates to the "List of Previous Feedback" page.

- Source: User (Teacher)
- Input: Accesses the page to view previous feedback.

2. Event: Teacher selects a course and date for filtering.

- Source: User (Teacher)
- Input: Chooses specific criteria for feedback filtering.

3. Event: Teacher clicks the "Filter" button.

- Source: User (Teacher)
- Input: Initiates the feedback filtering process.

Response

1. Event: System retrieves and displays the filtered feedback.

- Source: System
- Output: Fetches and shows feedback based on the selected criteria.

7.4. OTHER NON-FUNCTIONAL REQUIREMENTS

7.4.1. Performance Requirement

Visual elements should render smoothly for an immersive user experience. Minimum system requirements for mobile devices should be met.

7.4.2. Security Requirements

Data Encryption
All communication between the mobile app and external APIs (university authentication, ChatGPT, email server) should be encrypted using industry-standard protocols (e.g., HTTPS).

User Authentication
User authentication for teachers, students, and administrators should follow best practices. Passwords must be securely stored using hashing algorithms, and multi-factor authentication is recommended.

API Authentication
The system should implement secure authentication mechanisms when interacting with external APIs (university authentication, ChatGPT, email server). API keys and credentials must be securely stored.

7.4.3. Software Quality Attributes

Reliability
The system should be highly reliable, ensuring minimal downtime and consistent performance. Error handling mechanisms should be in place to manage unexpected situations gracefully.

Scalability
The architecture should be designed to handle a growing number of users and feedback requests. Scalability testing should be conducted to ensure optimal performance under varying loads.

Usability
The user interfaces for teachers, students, and administrators should be intuitive and user-friendly. Usability testing should be conducted to gather feedback and make improvements.

7.4.4. Safety Requirement

Data Privacy

The system must comply with data privacy regulations, ensuring that user information and feedback data are stored securely. Access controls should be in place to restrict unauthorized access to sensitive data.

System Monitoring

Implement a system monitoring solution to detect and respond to potential security threats or performance issues promptly. Regular audits of system logs should be conducted to ensure system integrity.

4. REFERENCES

[1] Çankaya University Official Website. https://cankaya.edu.tr

[2] https://openai.com/blog/introducing-chatgpt-and-whisper-apis