Bilkent University
Department of Computer Engineering

Senior Design Project
Team ID: T2510
Project Name: Mentora

Analysis and Requirement Report

Alper Biçer - 22003097 - alper.bicer@ug.bilkent.edu.tr
Kerem Demirören - 22203380 - kerem.demiroren@ug.bilkent.edu.tr
Gülferiz Bayar - 21901442 - gulferiz.bayar@ug.bilkent.edu.tr
Farabi Sina Sari - 22102084 - sina.sari@ug.bilkent.edu.tr
Berfin Örtülü - 21802704 - berfin.ortulu@ug.bilkent.edu.tr

Supervisor: Uğur Güdükbay
Course Instructors: Mert Bıçakçı, İlker Burak Kurt

17/12/2025

This report is submitted to the Department of Computer Engineering of Bilkent University in partial fulfilment of the requirements of the Senior Design Project course CS491/2.

Contents

Analysis and Requirement Report

Project Short-Name: MENTORA

1 Introduction

This report presents the analysis and requirements of Mentora, a mobile-based personalized study assistant developed within the scope of the CS 491 Senior Design Project. Mentora is designed primarily for students who require structured study planning, motivation support, and effective time management; however, its functionality is flexible enough to be used by other individuals seeking organized and adaptive study routines.

Mentora is intended to enhance study efficiency by providing users with a week- and day-long study plan according to their academic details, personality parameters, and emotional responses towards study. Emotional state perception is performed using a third-party, text-based emotion detection model, while study plan adaptation is carried out through mathematically formulated and deterministic rules. The system is intended as a productivity and study support tool and does not provide psychological diagnosis or therapeutic guidance.

The goal of this report is the analysis of the current state of study planning systems, the description of the proposed Mentora system, and the definition of the functional and non-functional requirements that determine the behavioral and quality characteristics of the latter. Moreover, high-level pseudo requirements are included that define the principal assumptions of the implementation. This report will be the base of subsequent design and implementation activities of the project.

2 Current System

In the present day, it seems that the majority of students use computer-based study management and productivity tools. These systems generally provide basic functionalities such as task lists, calendars, Pomodoro timers, reminders, and progress tracking. While such tools help users maintain a certain level of structure, most existing solutions remain largely static and offer limited personalization beyond simple scheduling preferences.

There are several popular applications which demonstrate the existing development level of study planning systems. MyStudyLife and Smart Timetable provide classic academic planners that focus on timetable management, assignment tracking, notifications, and Pomodoro-based study sessions. These applications are effective in organizing academic information; however, they typically apply the same planning logic to all users and do not adapt schedules based on individual characteristics such as emotional state or stress level.

More advanced platforms attempt to incorporate intelligent features. Smart Study AI concentrates on creating the content by using the artificial intelligence aspect to create questions, flashcards, and summaries from the studied material uploaded by the user. While this approach supports active learning, it mainly concentrates on content preparation rather than the management of time and workloads. Vaia integrates the aspects of learning schedules, chatbots, streaks, and collaborative learning content, providing the learner with a more advanced interface experience. However, personalization is mostly limited to predefined patterns and content recommendations.

Evergreen, a university-developed platform, places stronger emphasis on emotional awareness, behavioral knowledge, and student well-being. Even though Emotion Insight introduces considerations on emotions, their realm is wider and centered on support and well-being, rather than detailed adaptation in their learning plans depending on academic and stress variations throughout the day.

Overall, existing systems either emphasize organizational tools without deep personalization or focus on AI-driven content support. However, they often fail to integrate emotional feedback into structured study planning. Most applications also lack a clear systematic approach in terms of adapting study plans according to academic requirements as well as the emotional state of the users. Therefore, students tend to compensate manually by adapting their plans, reducing study intensity, or abandoning schedules during periods of stress or low motivation.

These limitations highlight the need for a system that balances academic planning with emotional awareness and adaptive behavior in a controlled and transparent manner. This gap in current solutions forms the basis for the proposed Mentora system, which aims to provide adaptive study planning through mathematically formulated rules while integrating emotional perception as supportive input.

3 Proposed System

This section presents the proposed solution developed as a result of the analysis of existing study planning systems and their limitations. Based on the identified gaps in current solutions, Mentora is designed to provide a more adaptive, structured, and emotionally aware study planning experience when planning a student's studies. The proposed system is aimed at supporting users in maintaining consistent study habits by combining academic workload information with emotional feedback in a transparent and controlled manner.

Mentora addresses the shortcomings of current systems by introducing adaptive study planning which responds to a change in the level of stress and learning context, while avoiding unclear or uncontrolled decision-making. In this system, decisions are based on mathematical expressions of rules rather than artificial intelligence for predictability and equity among users. Details of the system's design and operation are presented in the following overview.

3.1 Overview

Mentora is a mobile personalized learning assistant for studying, which is targeted mainly for students but can also be used by anyone who wants to have a systematic and adaptive learning schedule. The application is developed as a cross-platform mobile application using a framework called React Native, and the backend is developed using FastAPI. The application architecture uses a client-server architecture with RESTful communication to perform user interactions, analysis, and management of tasks on the server side.

Users interact with Mentora by creating a profile and entering academic information such as courses, exams, and assignment deadlines. In addition, users provide short, text-based daily feedback describing their emotional state.The data is analyzed through an external, third-party emotion detection model based on DistilRoBERTa, which classifies English text into predefined emotion categories and returns confidence scores. This component is used exclusively for emotional state perception and does not directly control planning decisions.

A numerical stress level is calculated based on the emotional state that has been identified and the amount of academic work that the user has. Stress values are normalized using mathematically defined calculations to ensure consistency among users. The adaptation of study plans is performed using formulated and deterministic rules, rather than AI-driven decision-making mechanisms. These rules regulate study duration, break intervals, task prioritization, and workload distribution in a transparent manner.

Mentora creates structured weekly study plans that are further divided into daily task schedules. During execution, the system tracks study progress, completed tasks, and total study time. When changes in stress level or academic context are detected, the system updates the study plan according to the predefined formulas, allowing controlled and predictable adaptation.

Besides planning and tracking, it also has study sessions, as well as recommended times to take a break, all based on the Pomodoro techniques. The system also includes an application-level chatbot that uses artificial intelligence only for navigational and informational purposes, such as guiding users to features within the application. The chatbot does not provide any tutoring or any psychological insights.

Overall, the proposed system combines emotional awareness, academic structure, and adaptive planning into one system. By clearly separating emotion perception and user guidance from deterministic planning logic, Mentora offers a balanced approach that enhances personalization while maintaining reliability and interpretability.

3.2 Functional Requirements

This section defines the functional requirements of the Mentora system. Functional requirements specify the observable behavior of the system and describe the fundamental processes and transformations performed on user inputs to produce meaningful outputs. These requirements focus on what the system must do in order to support personalized, adaptive, and structured study planning.

To improve clarity, the functional requirements are grouped into system functionalities and user functionalities, reflecting the responsibilities of the system and the actions available to users.

3.2.1 System Functionalities

The system shall be responsible for storing information related to the studies of users. Information can include courses, exams, deadlines for assignment submission, and study time. Based on this information, the system shall generate structured study schedules on a weekly basis and decompose them into daily task schedules.

The system shall analyze users' text-based emotional feedback using an externally integrated, third-party emotion detection model. The detected emotional state and associated confidence values shall be used as input parameters for further processing and shall not directly trigger autonomous decision-making.

The system shall calculate a numerical value for stress based on workload factors and emotional perceptions. Stress values shall be normalized using mathematically formulated calculations to ensure consistent interpretation across users.

The system shall adapt study plans based on deterministic and mathematically formulated rules created from the calculated stress level and academic context. These adaptations shall include adjustments to study duration, break intervals, task prioritization, and allocation of work.

The system shall track users' study progress by monitoring completed tasks, total study time, and adherence to planned schedules. Based on this tracking, the system shall modify the future schedule when predefined conditions are met.

The system shall support Pomodoro study sessions and automatically manage study and break cycles according to the generated plan. Break activity recommendations may be modified depending upon the stress levels of the users.

The system shall provide notifications related to upcoming tasks, deadlines, and study reminders in a timely manner. Additionally, the system shall generate supportive feedback messages aimed at encouraging successful study habits.

The system shall provide an application-level chatbot that assists users with navigation and feature discovery. This chatbot shall be limited to informational and guidance purposes and shall not provide academic or psychological advice.

3.2.2 User Functionalities

Users shall be able to create a personal account and securely access the Mentora application. Users shall enter and update academic information such as enrolled courses, exams, assignments, and preferred study hours.

Users shall provide daily or periodic text-based feedback describing their emotional state. Users shall also be able to review their generated weekly and daily study plans within the application.

Users shall be able to start, pause, and stop Pomodoro-based study sessions and follow system-recommended break periods. Users shall track their study progress, completed tasks, and accumulated study time through visual summaries.

Users shall receive notifications and feedback generated by the system and may adjust certain personal preferences, such as notification settings and visibility of performance comparisons. Users shall interact with the in-application chatbot to locate features and learn how to use system functionalities.

3.3 Non-functional Requirements

This section defines the non-functional requirements of the Mentora system. Non-functional requirements describe quality attributes and constraints related to system performance, usability, reliability, maintainability, scalability, and security. These requirements specify how the system should operate rather than what it should do.

3.3.1 Performance and Efficiency

The Mentora system shall respond to user interactions without noticeable delay. Study plan generation, stress level computation, and emotion analysis results shall be delivered within a reasonable time to preserve a smooth user experience. Pomodoro timers and notifications shall operate accurately and

on time, without causing performance degradation on the mobile device. Visual elements and animations shall not negatively affect application responsiveness.

3.3.2 Usability

The system shall provide a simple and intuitive user interface suitable for mobile devices with different screen sizes. Users shall be able to understand system functionalities and navigate between features without extensive learning effort. The interaction flow shall remain consistent throughout the application. Feedback messages, notifications, and visual cues shall be designed to support users without creating distraction or cognitive overload.

3.3.3 Reliability and Dependability

The system shall reliably store and retrieve user data, including academic information, study plans, progress records, and emotional feedback. User data shall not be lost during normal operation, application restarts, or temporary connectivity issues. Core functionalities such as study plan access, progress tracking, and Pomodoro timing shall operate consistently and predictably.

3.3.4 Maintainability and Supportability

The Mentora system shall be designed in a modular and organized manner to support future maintenance and updates. Backend and frontend components shall remain loosely coupled to allow independent modification. The system shall support the extension or replacement of components such as emotion detection models or planning rules without requiring major architectural changes.

3.3.5 Scalability

The system shall continue to function correctly as the number of users and stored data increases. Increased user activity shall not lead to significant performance loss. If required, resource-intensive components such as emotion analysis may be handled as separate services to maintain system responsiveness.

3.3.6 Privacy and Security

The system shall protect user data against unauthorized access. User authentication data shall be securely stored using appropriate protection mechanisms. Communication between the mobile client and backend services shall be secured. Emotional feedback, personality-related information, and academic data shall be treated as sensitive personal data and processed only with explicit user consent. Users shall be able to view, update, or request the removal of their personal data in accordance with applicable data protection regulations.

3.4 Pseudo Requirements

● The Mentora system will be developed as a cross-platform mobile application using React Native.

● The backend services of the system will be implemented using the FastAPI framework in Python.

● Communication between the mobile client and backend services will be performed through RESTful APIs using structured data formats.

● A relational database system will be used to store user accounts, academic information, study plans, emotional feedback, and progress data.

● Third-party, pre-trained natural language processing models will be integrated to perform emotion detection on English text inputs.

● Emotion detection models will be used only for emotional state perception and will not perform study plan generation or autonomous decision-making.

● Study plan adaptation and stress-based adjustments will be implemented using mathematically formulated and deterministic rules, rather than AI-driven planning mechanisms.

● An application-level chatbot component will be included to assist users with navigation and feature discovery within the application.

● Development and testing activities will be carried out in local environments using personal or university-provided machines.

● The system will not depend on commercial cloud services or paid third-party platforms as part of its baseline design.

● Git and GitHub will be used as the version control system to support collaborative development and traceability.

3.5 System Models

3.5.1 Scenarios

3.5.1.1 The Personalized Kickoff (Onboarding)

User: Alex, a first-year university student who feels overwhelmed by the transition from high school.

- The Action: After downloading Mentora, Alex completes the initial five-trait personality assessment. The results indicate that Alex scores high in Extraversion but moderate in Conscientiousness.
- The System Response: Mentora recognizes that Alex might struggle with isolation during long study sessions. It adjusts his study plan to include "Social Study Sprints" and suggests he shares his milestones on social media early on to build external accountability.
- The Outcome: Alex feels understood by the app immediately, as the suggested study blocks are shorter and more frequent, matching his personality profile rather than a generic template.

3.5.1.2 The Emotional Pivot (Daily Adjustment)

User: Sarah, a medical student who is feeling burnt out and anxious before a major exam.

- The Action: Upon opening the app, Sarah responds to the daily prompt ("How are you feeling today?") with: "I'm exhausted and I don't think I can get through this chapter."
- The System Response: Mentora's sentiment analysis detects high stress and low energy. Instead of the planned 4-hour intensive review, the app pivots: it suggests a "Low-Intensity Review" and incorporates two 30-minute mindfulness breaks.
- The Outcome: Because the plan adapted to her emotional state, Sarah completes the modified tasks instead of giving up entirely, maintaining her momentum without increasing her burnout..

3.5.1.3 The Academic Crunch (Deadline Management)

User: Jordan, a high school senior balancing multiple science classes and a part-time job.

- The Action: Jordan syncs his academic calendar, revealing a history of lower grades in Mathematics and a heavy History essay due in three days.
- The System Response: Mentora prioritizes Mathematics review in his daily slots to bolster his weak spots. It also automatically "backloads" the History essay, breaking it into smaller chunks (Outline, Draft, Edit) over the next 72 hours.
- The Outcome: Jordan doesn't have to decide what to study; he simply follows the plan that has already calculated the most efficient path to improving his grades while meeting his deadlines.

3.5.1.4 The Motivation Loop (Retention & Social)

User: Leo, a competitive student who thrives on peer recognition.

- The Action: Leo finishes a 5-day study streak. He checks the Mentora Leaderboard and sees he has dropped to 5th place among his classmates.

- The System Response: Mentora sends a push notification: "You're on a 5-day fire streak, Leo! One more hour today will put you back in the Top 3. You've got this!" The Outcome: Motivated by the leaderboard and the personalized encouragement, Leo completes an extra study module. He then posts his 5-day streak to his Instagram story, gaining validation from his friends.

3.5.1.5 The "Recovery" Scenario (Handling Setbacks)

User: Maya, a college student who has a high Neuroticism score (sensitivity to stress) and just failed a midterm.

- The Action: Maya enters her latest grade into her academic history and, in the daily check-in, writes: "I failed my midterm. I feel like a failure and I want to quit."
- The System Response: Mentora recognizes the high-stress markers. Instead of pushing a "harder" study schedule to compensate for the grade, the app uses its Motivational Feedback engine. It sends a gentle message: "One grade doesn't define your journey, Maya. Let’s take a small win today." It clears the heavy tasks and sets a single, 20-minute "Review of Mistakes" session.
- The Outcome: By lowering the barrier to entry, Mentora prevents Maya from spiraling. She completes the small task, earns a "Resilience Badge," and feels a sense of control again.

3.5.1.6 The "Deep Work" Scenario (High Conscientiousness)

User: David, a postgraduate researcher with a high Conscientiousness score and a "Productive" emotional state.

- The Action: David logs in and types: "I'm feeling focused and ready to tackle the heavy lifting today."
- The System Response: Mentora identifies that David’s personality profile thrives on structure and high-intensity goals. It optimizes his plan for "Deep Work," scheduling two 90-minute uninterrupted blocks

with "Focus Mode" enabled (silencing app notifications). It also updates his leaderboard stats to reflect "Time Spent in Deep Work," a metric David finds highly rewarding.

- The Outcome: David maximizes his peak productivity hours, and the app's data-driven feedback confirms he is working at his highest efficiency.

3.5.2 Use-Case Model

3.5.2.1 Profile & Personalization

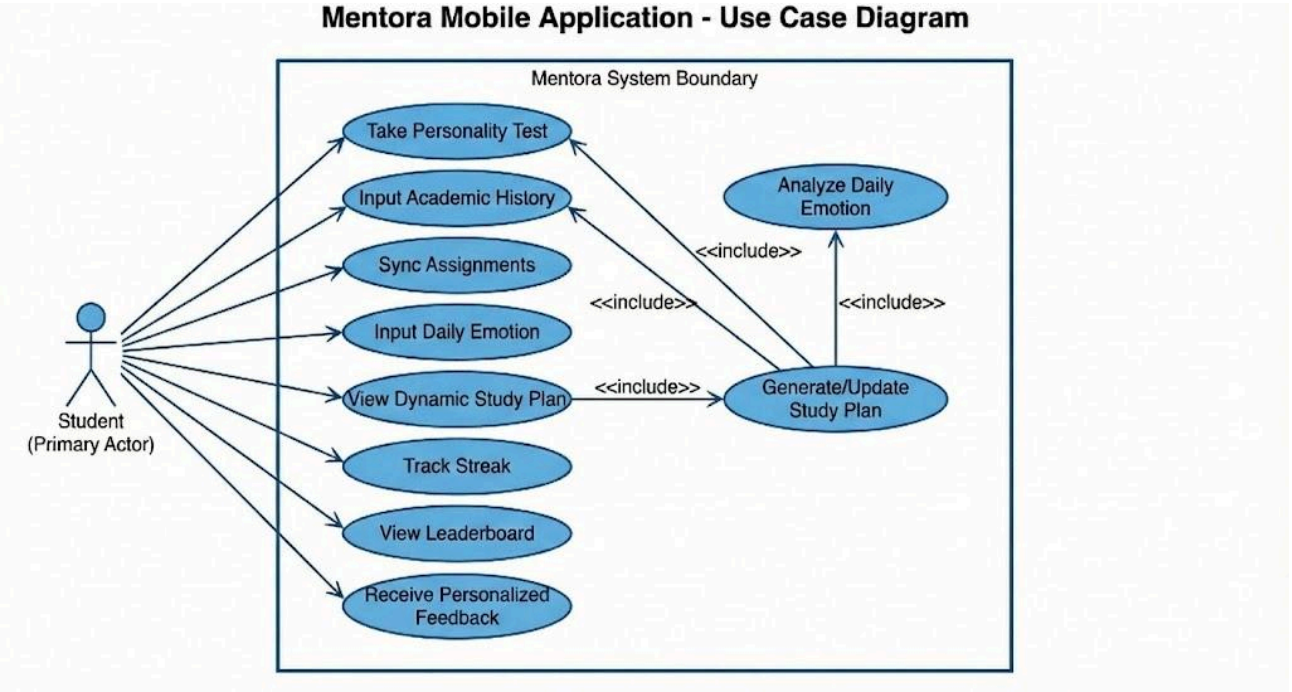
- Take Personality Test: The user completes the 5-trait test. The system stores these traits as a baseline for all future plan adjustments.
- Set Academic History: The user inputs past grades and subjects, allowing the system to identify "weak spots" that require more study time.

3.5.2.2 Daily Dynamic Adjustment

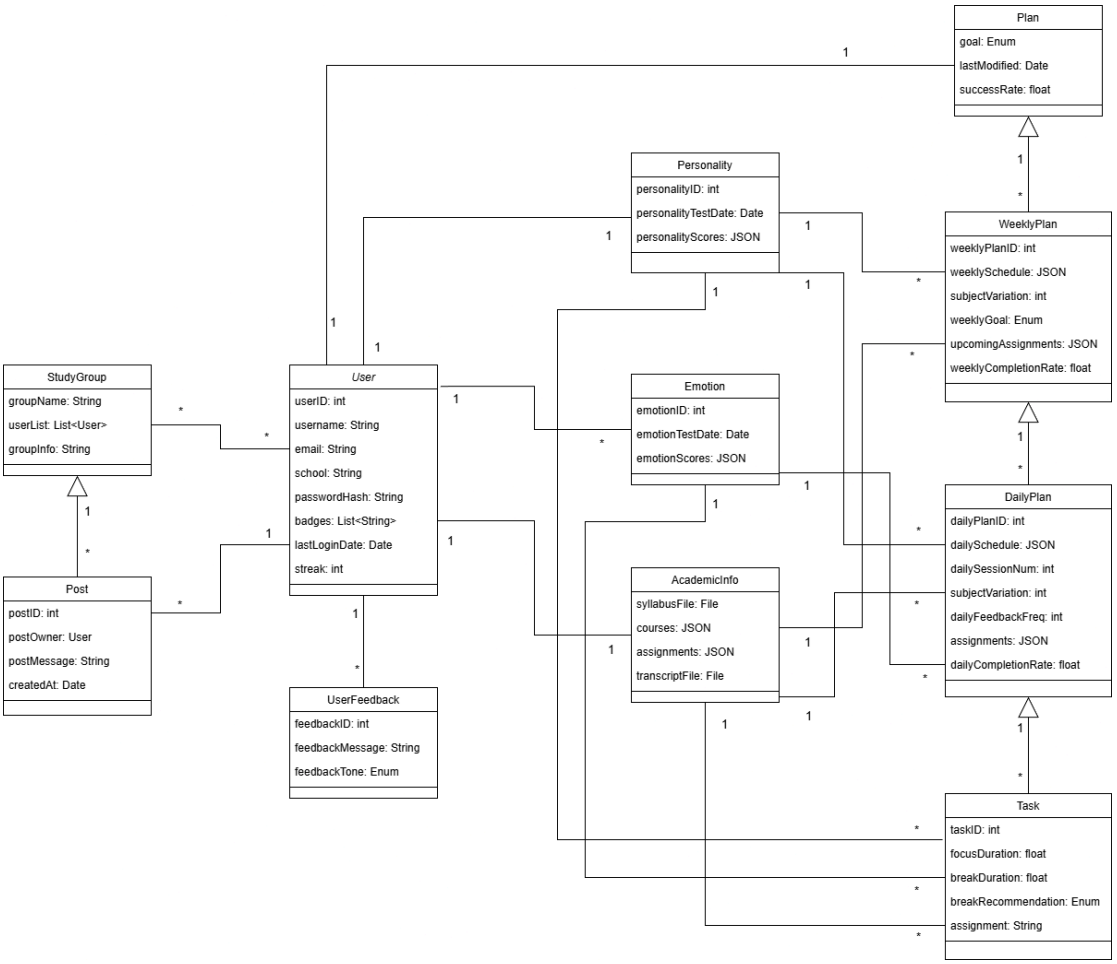
- Input Daily Emotion: The user provides a text-based response. The system uses Natural Language Processing (NLP) to categorize the sentiment.
- Generate/Update Study Plan: This is the core engine. It combines:
    - Personality traits (e.g., preference for social vs. solo study).
    - Current emotion (e.g., reducing workload if "stressed").
    - Upcoming assignments (e.g., prioritizing an essay due in 2 days).

3.5.2.3 Motivation & Gamification

- Track Streak: The system monitors consecutive days of usage to build a "habit loop."
- View Leaderboard: The user compares their progress with peers to encourage healthy competition.
- Receive Feedback/Motivational Messages: The system pushes notifications that are tailored to the user's personality (e.g., "gentle" for high neuroticism, "competitive" for high extraversion).



3.5.3 Object and Class Model



The User class stores the user profile information and also information that can be used for other features like streak and badges. Streak can be used to calculate how many consecutive days the user uses the application and badges can be used to allow users to complete achievements.

The user-specific information is held in separate classes. These classes are Personality, Emotion, AcademicInfo, and Plan. Each user has a single Personality, AcademicInfo, and Plan classes but can have many Emotion classes since emotions are gathered on a daily-basis.

The Plan class is inherited by WeeklyPlan. Weekly plans of the user are also set with the help of Personality and AcademicInfo classes. The WeeklyPlan class is inherited by DailyPlan. Daily plans of the user are set with the help of the Personality, Emotion, and AcademicInfo classes. The DailyPlan class is inherited by the Task class which represents a single task the user will do. Task class's specifications are calculated with the help of Personality, Emotion, and AcademicInfo classes.

Users can give feedback to the application which is held in a separate class. Also users can assign themselves to the study groups. The StudyGroup class stores the assigned users in a list. The StudyGroup class is inherited by the Post class. The Post class will be created for each post commented. The Post class inherits the StudyGroup class.

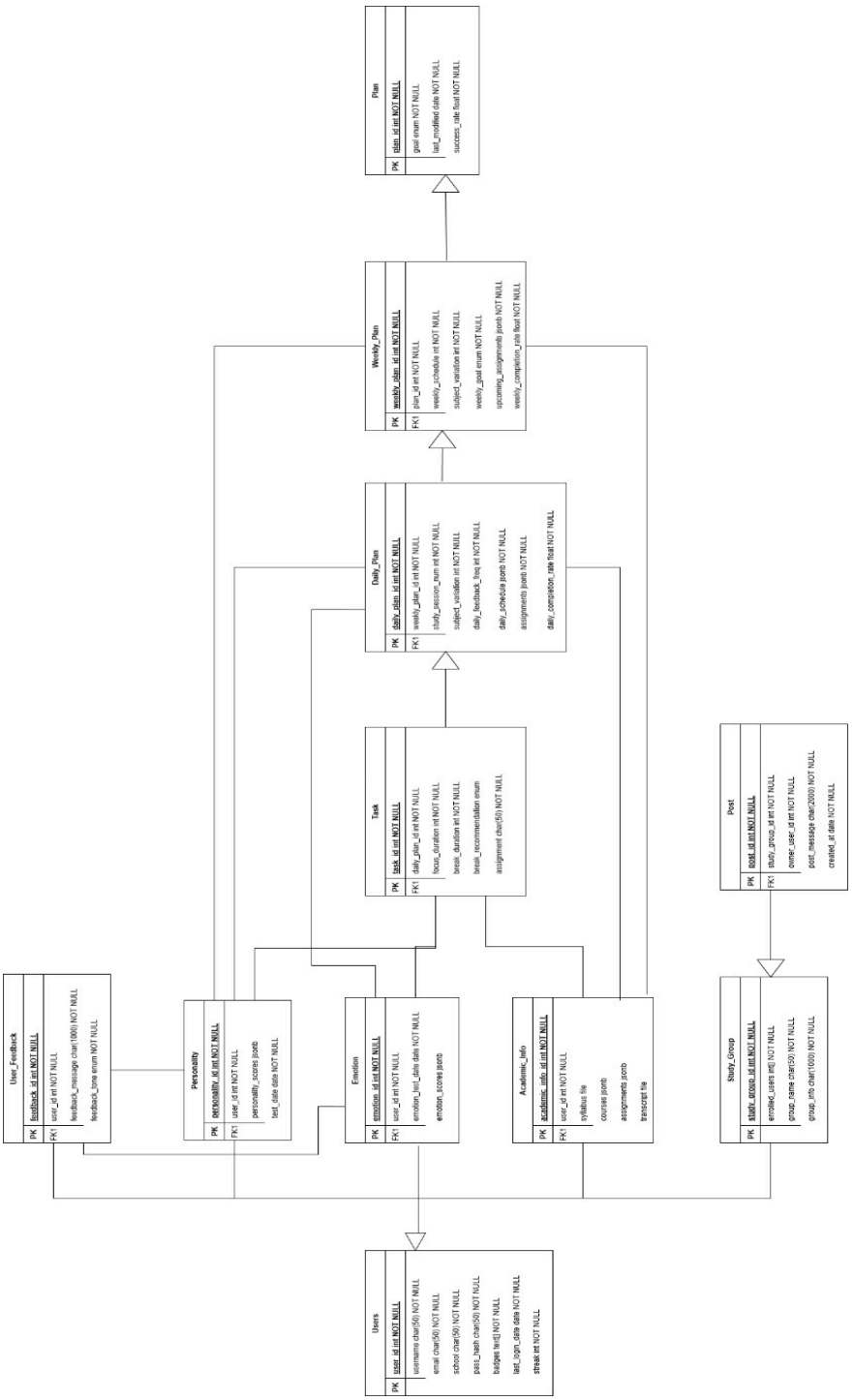
Table 1 is an example table.

Table 1. An example table.

Key	**Value**
key	Value

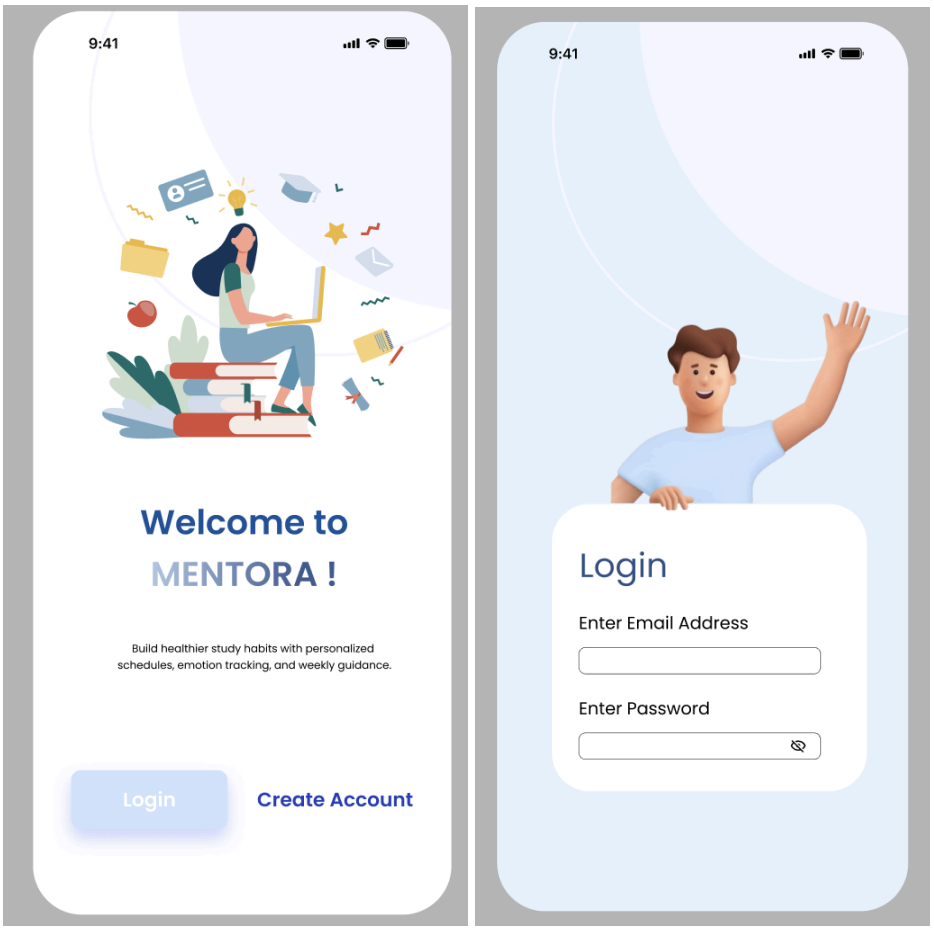
3.5.4 Dynamic Models

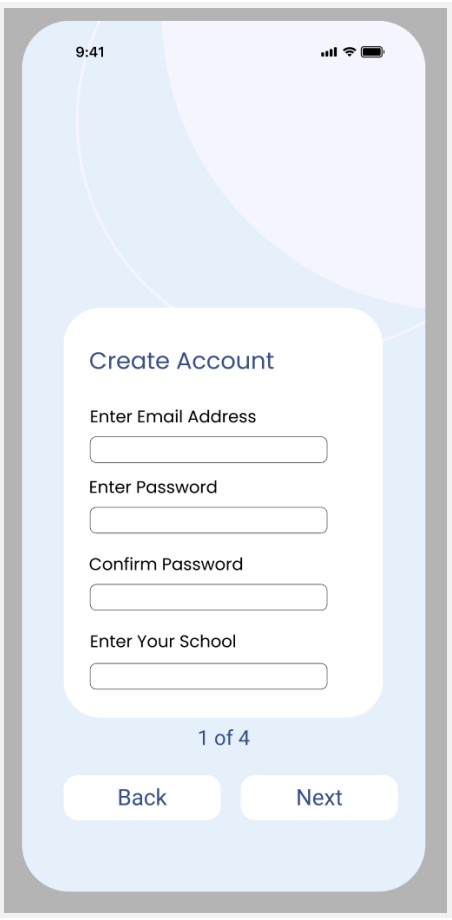
E-R Model

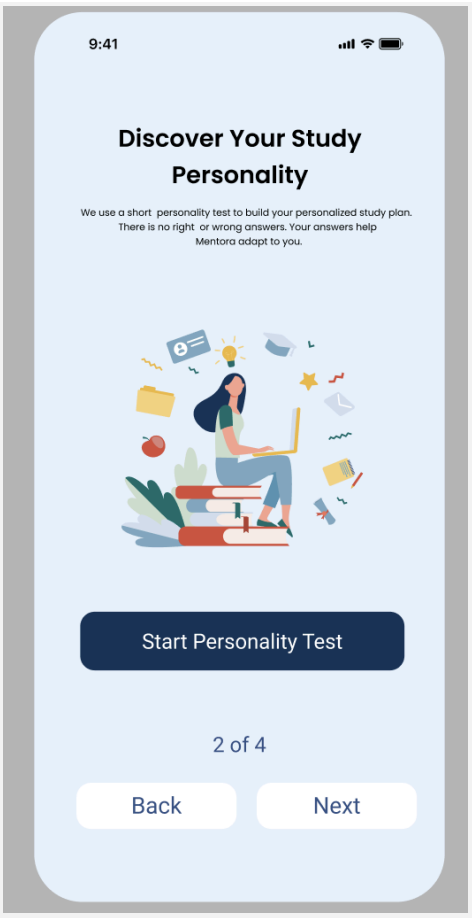


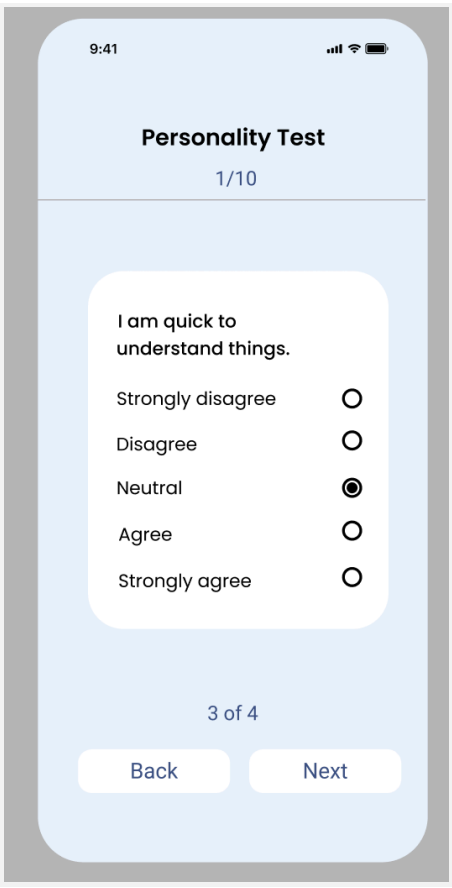
3.5.5 User Interface

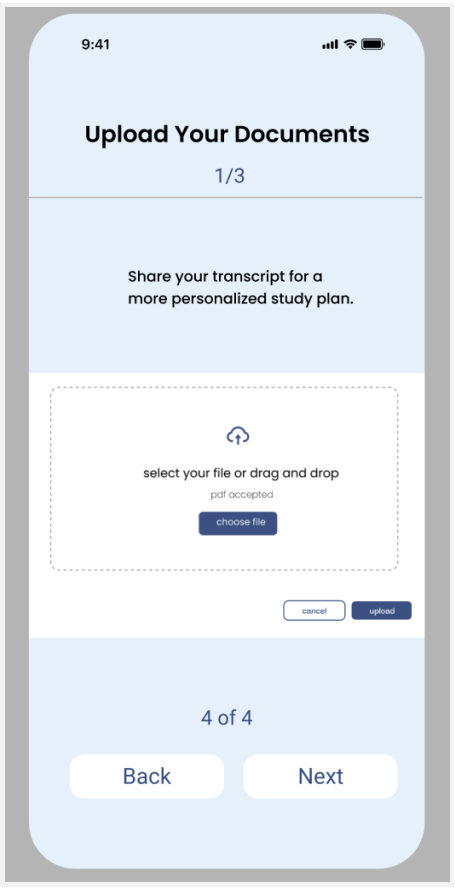
3.5.5.1 Login / Register

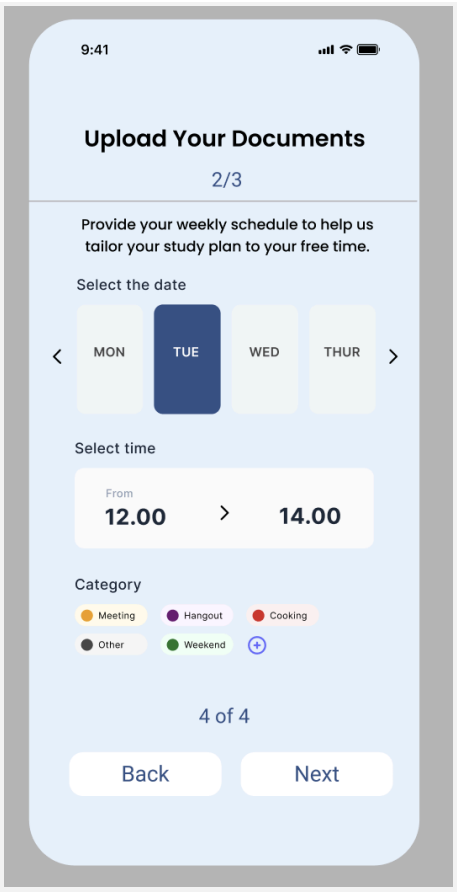



9:41
Create Account
Enter Email Address
Enter Password
Confirm Password
Enter Your School
1 of 4
Back
Next

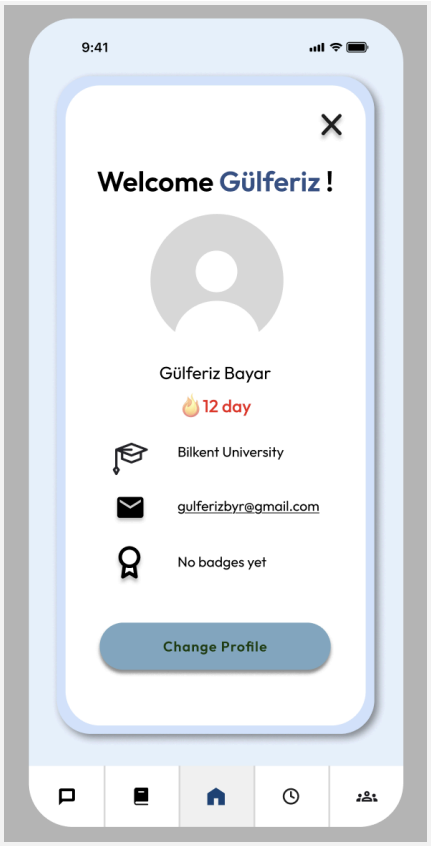

9:41
Discover Your Study Personality
We use a short personality test to build your personalized study plan. There is no right or wrong answers. Your answers help Mentora adapt to you.
Start Personality Test
2 of 4
Back
Next


9:41
Personality Test
1/10
I am quick to understand things.
Strongly disagree
Disagree
Neutral
Agree
Strongly agree
3 of 4
Back
Next

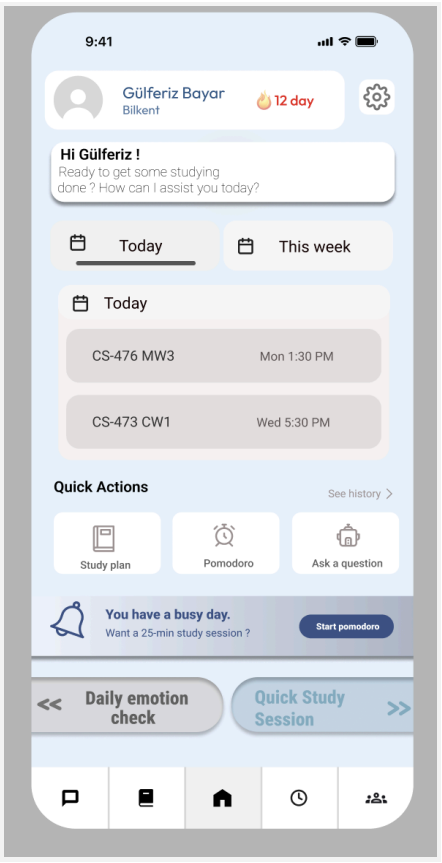

9:41
Upload Your Documents
1/3
Share your transcript for a more personalized study plan.
select your file or drag and drop
pdf accepted
choose file
cancel
upload
4 of 4
Back
Next



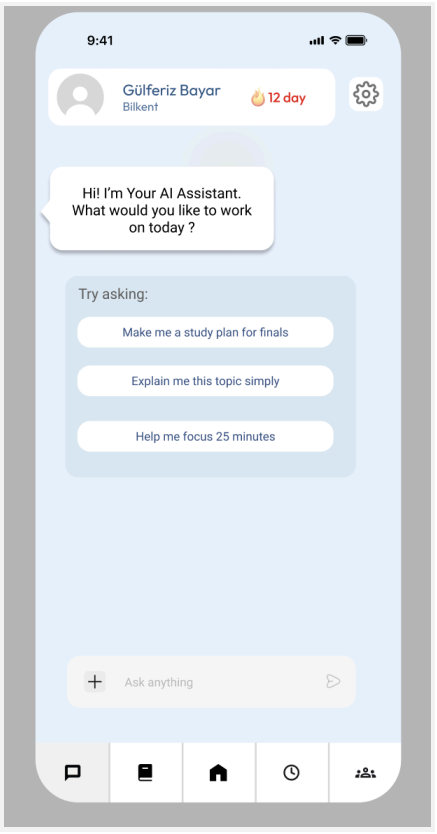
3.5.5.2 Pop Up Welcome Page



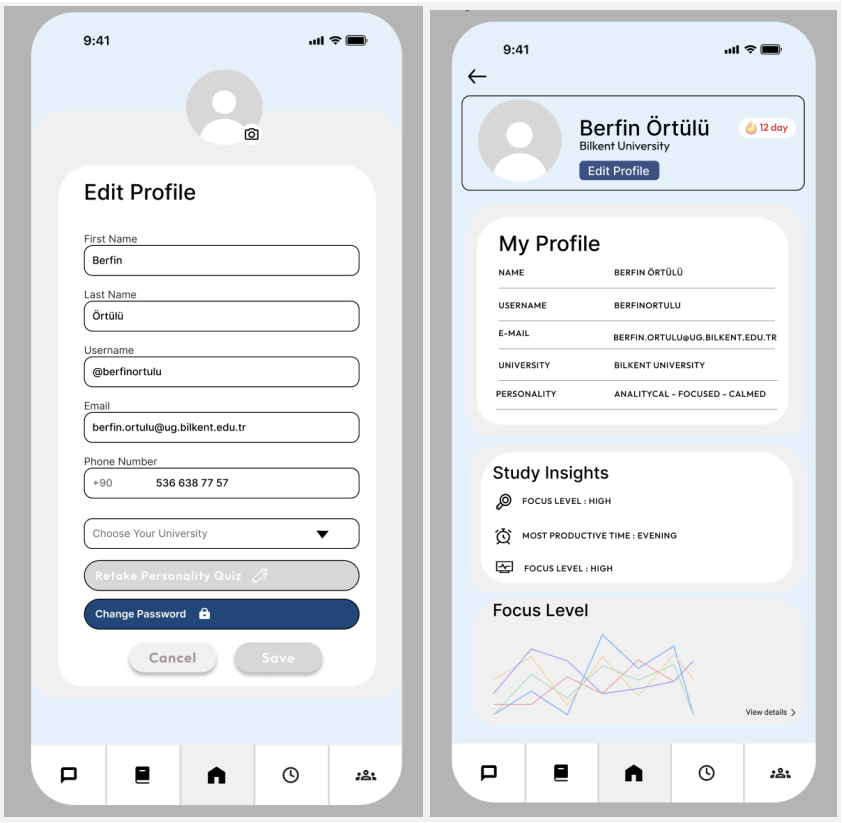
3.5.5.3 Home Page



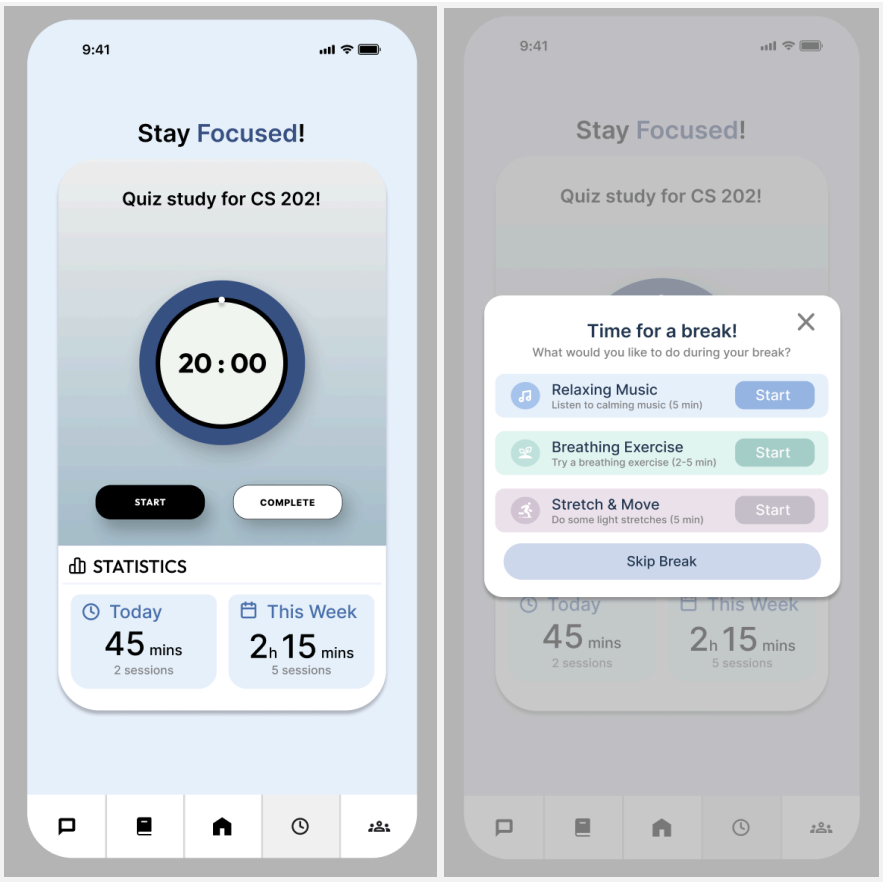
3.5.5.4 Chatbot Page



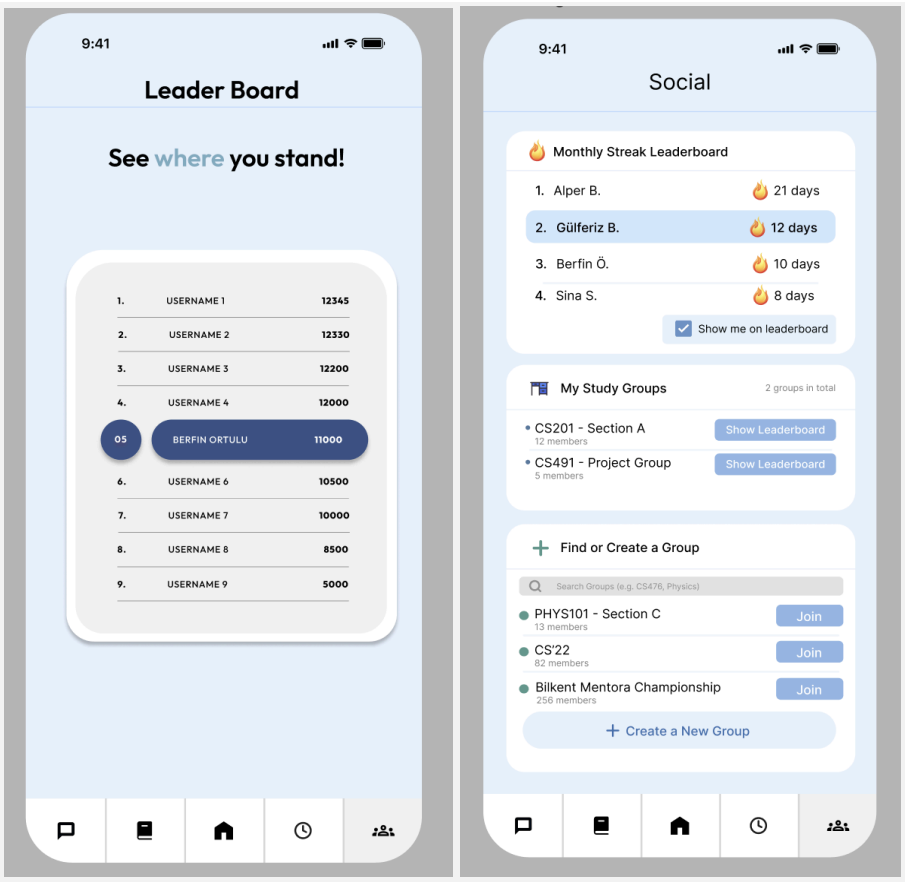
3.5.5.5 Profile Page



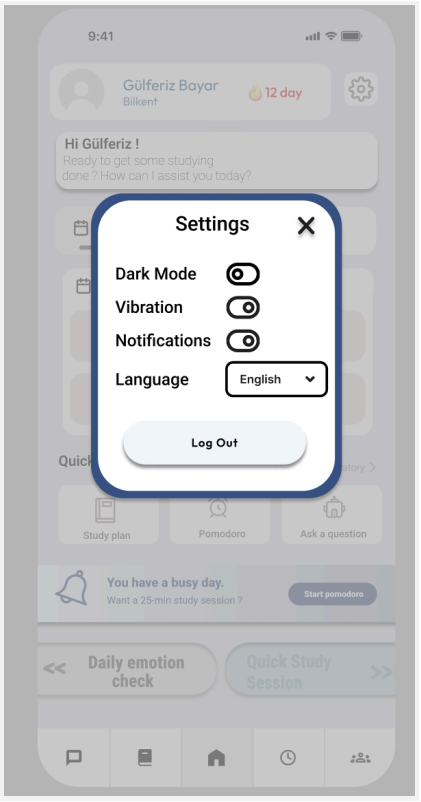
3.5.5.6 Pomodoro Page



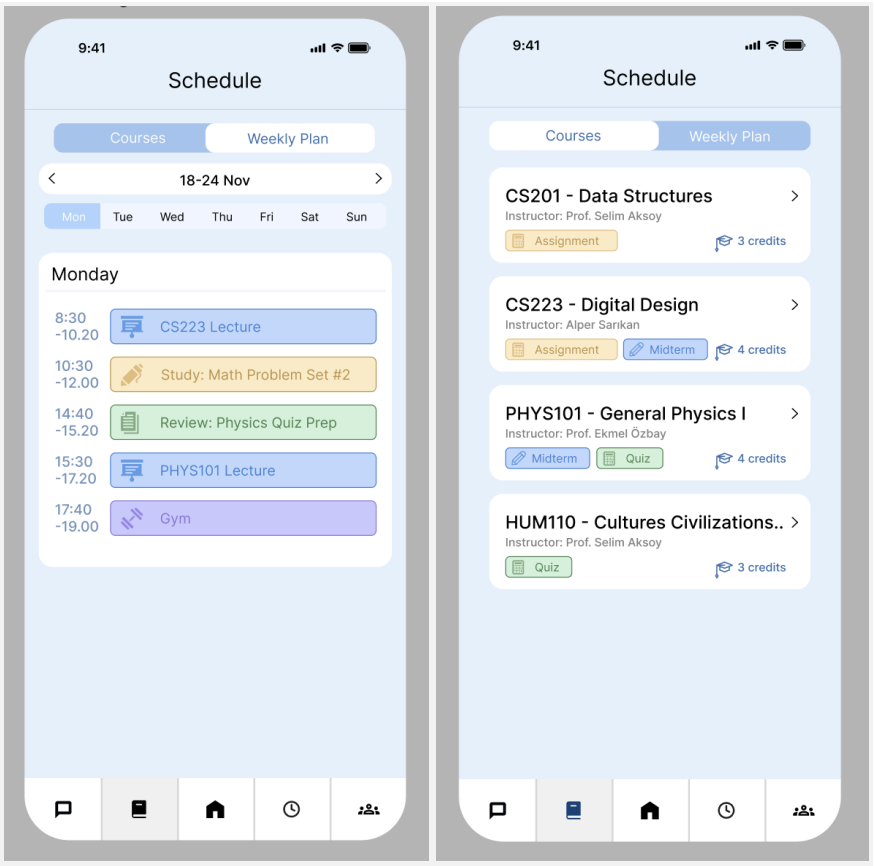
3.5.5.7 Leaderboard / Social Page



3.5.5.8 Settings Page



3.5.5.9 SCHEDULE PAGE



4 Other Analysis Elements

In this section, the main constraints that affect the design and implementation of Mentora are discussed. These include implementation, economic, ethical, and professional constraints, which directly influence architectural decisions, technology selection, and system behavior.

4.1 Consideration of Various Factors in Engineering Design

4.1.1 Constraints

4.1.1.1 Implementation Constraints

- Mentora will be implemented as a cross-platform mobile application.

- The frontend will be developed using Flutter; the same codebase will target at least Android and iOS mobile platforms.
- The backend will be implemented in Python using the FastAPI framework.
- Communication between the client and the backend will be done via a RESTful API over HTTP/HTTPS.
- A PostgreSQL relational database will be used as the main data store.
- SQLAlchemy will be used as the Object Relational Mapping layer to integrate FastAPI with PostgreSQL.
- The database must allow storing JSON fields for flexible data types such as personality test responses and weekly feedback forms.
- All main features of Mentora (creating study plans, Pomodoro timers, personality scoring, workload analysis and weekly updates) will run on the backend server, so the mobile application stays simple and lightweight.
- The emotion detection module will be implemented using open-source text-based AI models that can run within the Python/FastAPI environment.
- Text-based analysis of students' feedback will rely on open source NLP libraries compatible with Python.
- Source code will be developed using VS Code with formatting and linting extensions to maintain a consistent code style.
- Git and GitHub will be used for version control, branching, pull requests and issue tracking throughout the project.
- Only free or university provided tools and services will be used. No commercial SDKs or paid cloud services are assumed in the baseline design.

4.1.1.2 Economic Constraints

- Since the project doesn't have enough budget, all technologies that will be used must be open source and free.
- PostgreSQL, FastAPI, and Flutter are free options for this project because they offer complete functionality without requiring a payment.
- For version control, teamwork, and issue tracking, GitHub's free plan will be used.
- To avoid the high cost of commercial APIs, open-source libraries will be used to implement NLP and emotion detection modules.
- Because of the financial constraints, development and testing will be done using personal laptops or computers provided by the university; large-scale cloud services cannot be used.
- Since paid datasets are not practical for the project, all datasets required for development will be sourced from free internet sources.

4.1.1.3 Ethical Constraints

- Mentora processes sensitive personal data like academic history, personality test scores, daily emotional situation, and weekly reflections. These data types must be carefully handled due to their sensitive nature.
- By using the data minimization principle, only the minimum amount of data needed for personalization will be gathered from the user.
- Users must be informed about all data processing activities with a privacy notice and an explicit consent form, specifying data categories, usage purposes, storage durations, and user rights on these data.
- Emotional data and personality profiles gathered from the user will be handled with caution and stored only when required, based on discussions with psychology and law faculty members [3–4].

- To prevent unauthorized access, the system will use technical safeguards like hashed and salted password storage, and encrypted communication between client and server.
- Users must be able to access, update, or request the deletion of their personal data within reasonable limits.
- The system must avoid any design decisions that could result in psychological pressure on the users. Feedback and recommendations to the users should remain supportive and within mental health principles.
- No user data will be shared with third-party services unless explicitly stated and approved through a written agreement with the user.
- The project will follow the ethical guidance provided during our faculty consultations, while also complying with broader privacy and legal standards such as KVKK.

Table 2: Factors that affect the analysis and design decisions of the Mentora system.

Factor	Effect Level	Effect on Mentora Design
Public Health	2	The system avoids intrusive or stressful interactions and promotes healthy study habits through breaks and well-being suggestions.
Public Safety	1	Mentora is a software-based system and does not pose physical safety risks to users.
Public Welfare	1	The system indirectly supports student well-being by improving study organization and reducing academic stress.
Global Factors	2	Mentora is designed to be language-independent and adaptable to different academic calendars and education systems.
Cultural Factors	2	Study recommendations are personalized based on individual behavior rather than cultural assumptions.

Social Factors	4	The system must remain supportive, non-judgmental, and non-intrusive to avoid increasing social or academic pressure.
Environmental Factors	1	As a software-only application, Mentora has minimal environmental impact.
Economic Factors	5	Budget limitations strongly influence technology choices, requiring the use of free and open-source tools.
Ethical Factors	6	Processing emotional and personality data requires strong privacy, consent, and data protection mechanisms.

4.1.2 Standards

- The structure and behavior of the system will be modeled using UML 2.5.1 standard. This standard includes diagrams that will help to clearly and consistently represent the design, such as use case, class, and activity diagrams [5].
- The project's requirements will be documented by following the IEEE 830 standard, guaranteeing that all functional and non-functional requirements are written in an intelligible, structured, and clear manner [6].
- We will do our work during the development process with the general framework offered by IEEE 12207 standard, which specifies the phases of the software life cycle, including planning, analysis, design, implementation, testing, and maintenance [7].
- To have an organized and traceable design process, design-based decisions like layout, module descriptions, and interface definitions will be done using the IEEE 1016 standard [8].
- We will use IEEE 1058 standard, which provides helpful methods for planning and overseeing software projects, as a guide for project planning and management tasks [9].

4.2 Risks and Alternatives

One of the main risks in the Mentora project is the incorrect interpretation of emotional or personality related data. Emotion detection and personality profiling are inherently uncertain and may not always reflect the user's actual mental state. This could lead to less effective or inappropriate study recommendations. To reduce this risk, Mentora avoids making strong claims and presents its suggestions as supportive guidance rather than definitive conclusions.

Data security and privacy represent another significant risk. Any data breach or misuse could have a detrimental impact on users because Mentora handles sensitive personal information like personality traits, academic history, and emotional states. Applying encryption, secure authentication procedures, robust access control, and gathering only the bare minimum of information are ways to mitigate this risk.

Another risk would be in relation to the involvement and motivation of the users. There may be some users who no longer use the system or those who provide inaccurate feedback, leading to decreased efficiency by the personalization system. Another feature would be for Mentora to allow users to manually modify their study plans and preferences.

On the other hand, there is also a risk of technical limitation because of the constraint imposed on the use of the computer resources and the use of open-source software tools. Advanced or complex artificial intelligence models may not be suitable or even feasible for execution efficiently. Another approach to this problem is the use of simple logic and machine learning tools by the Mentora project.

Finally, as an alternative design choice, Mentora could be implemented without emotion detection and rely only on user-declared inputs and academic data. While this approach would reduce privacy concerns and technical complexity, it would also limit the level of personalization. Therefore, the current hybrid approach is preferred.

Table 3: Identified risks in the Mentora project and corresponding alternative solutions.

Risk Name	Likelihood	Effect on the Project	Alternative / B Plan Summary
Incorrect interpretation of emotional or personality data	Medium	Ineffective or inappropriate study recommendations may be generated	Use emotion analysis only as supportive input, rely on user feedback and avoid definitive conclusions
Data privacy and security breach	Low–Medium	Loss of user trust and potential legal issues	Apply encryption, secure authentication, access control, and data minimization principles
Low user engagement or inaccurate feedback	Medium	Reduced personalization quality and system effectiveness	Allow users to manually adjust study plans and preferences
Technical limitations of AI models	Medium	Advanced AI models may not run efficiently or reliably	Use lightweight rule-based logic and simple machine learning techniques
Over-reliance on emotion detection	Low	Increased ethical and privacy concerns	Provide an alternative mode without emotion detection, relying only on academic data

4.3 Project Plan

Table 4: List of work packages

WP#	Work package title	Leader	Members involved
WP1	Project Idea Generation and Initial Research	Gülferiz	All members
WP2	Project Specifications Report	Kerem	All members
WP3	Consultations with Authorized Faculty Members	Alper	All members
WP4	Analysis and Requirement Report	Berfin	All members
WP5	User Interface (UI) & Backend Design	Sina	All members

Table 5 : Gantt chart for the work packages

WP	Work package title	Sep 2025		Oct 2025		Nov 2025		Dec 2025	
WP1	Project Idea Generation and Initial Research								
WP2	Project Specifications Report								
WP3	Consultations with Authorized Faculty Members								
WP4	Analysis and Requirement Report								
WP5	User Interface (UI) & Backend Design								

Table 6: Detailed description of work packages

WP 1: Project Idea Generation and Initial Research			
Start date: *20.09.2025* **End date:** *25.09.2025*			
Leader	*Gülferiz*	**Members involved:**	*All members*
Objectives: The main objective of this work package is to define the Mentora project idea clearly and build a strong conceptual foundation for the system. During this phase, the team aims to identify the core problems students face in their study routines and explore how a personalized study assistant can address these challenges. Initial research is conducted on related systems, personalization approaches, and potential technologies to ensure that the project idea is feasible and meaningful. This work package also aims to align the project scope with academic, ethical, and practical constraints.			
Tasks: ***Task 1.1*** **Brainstorming and Project Idea Definition:** The team brainstorms different project ideas and discusses potential problem domains. Based on these discussions, the Mentora concept is selected and its main goals, target users, and core features are defined. ***Task 1.2*** **Literature Review and Related Work Analysis:** Existing studies, applications, and systems related to study planning, personalization, emotion-aware systems, and productivity tools are researched. This task helps our team to understand current approaches and identify gaps that Mentora can address. ***Task 1.3*** **Feasibility and Scope Analysis:** The feasibility of the proposed project idea is evaluated in terms of technical requirements, available resources, and			

time constraints. The project scope is refined to ensure that the planned features can be realistically implemented within the course timeline.

Task 1.4 **Initial Technology and Methodology Exploration:** Possible technologies, frameworks, and methodologies are explored at a high level. This includes preliminary discussions on frontend, backend, database technologies, and the use of AI-based components.

Deliverables
D1.1: *Project Information Form*
D1.2: *Assessment of Innovation Form*

WP 2: Project Specifications Report			
Start date: *16.11.2025* **End date:** *27.11.2025*			
Leader	*Kerem*	**Members involved:**	*All members*

Objectives: The objective of this work package is to prepare the Project Specifications Report for the Mentora system. This report aims to clearly describe the proposed solution, its system architecture, and its main components. During this phase, functional and non-functional requirements are defined in a structured manner. In addition, project constraints, professional and ethical responsibilities, and engineering standards relevant to Mentora are documented in detail to ensure a clear and well-justified system definition.

Task 2.1 **System Description and High-Level Architecture Definition:** This task focuses on describing the Mentora system, its purpose, and its main components. A high-level architecture is defined, including frontend, backend, database, and AI-related modules.

Task 2.2 **Requirements Analysis and Documentation:** Functional and non-functional requirements of the system are identified and documented. These requirements are organized according to usability, reliability, performance, supportability, and scalability considerations.

Task 2.3 **Constraint Analysis:** Project constraints such as implementation, economic, ethical, social, and health-related constraints are analyzed and documented. This task ensures that the system design aligns with realistic limitations and responsibilities.

Task 2.4 **Professional, Ethical Issues and Standards Definition:** Professional and ethical responsibilities related to the project are discussed. In addition, engineering and documentation standards such as UML and IEEE standards are identified and explained.

Task 2.5 **Compilation and Review of the Project Specifications Report:** All sections of the report are compiled into a single document. The report is reviewed for consistency, clarity, and alignment with course guidelines before submission to the project supervisor and course instructors.

Deliverables
D2.1: *Project Specification Document*

WP 3: Consultations with Authorized Faculty Members			
Start date: *25.11.2025* **End date:** *26.11.2025*			
Leader	*Alper*	**Members involved:**	*All members*

Objectives: The objective of this work package is to support the Mentora project through consultations with authorized faculty members and domain experts. These consultations aim to guide the team on psychological, legal, ethical, and technical aspects of the system. By receiving expert feedback, the team ensures that design decisions, data usage, and personalization mechanisms are aligned with academic, ethical, and legal standards. This work package also helps improve the reliability and responsibility of the proposed solution.

Tasks:

Task 3.1 **Psychology-Oriented Consultations:** Meetings are conducted with psychology faculty members to discuss personality assessment methods, emotion interpretation, and the presentation of emotional feedback to users. These discussions guide how Mentora adapts study plans without causing psychological pressure.

Task 3.2 **Legal and Ethical Consultations:** Consultations with law faculty members focus on data protection regulations such as KVKK and GDPR, consent requirements, and user rights. This task ensures that the system design complies with legal and ethical obligations when handling sensitive personal data.

Task 3.3 **Technical and Innovation-Oriented Feedback Sessions:** Discussions are held with faculty members and experts experienced in system design and innovation. Feedback received during these sessions is used to refine system architecture, feature prioritization, and overall feasibility.

Task 3.4 **Documentation and Integration of Feedback:** Key outcomes from all consultations are documented and incorporated into system requirements, constraints, and design decisions to ensure consistency across project deliverables.

Deliverables
No deliverables

WP 4: Analysis and Requirement Report			
Start date: *09.12.2025* **End date:** *19.12.2025*			
Leader	*Berfin*	**Members involved:**	*All members*

Objectives: The objective of this work package is to perform a detailed analysis of the Mentora system and to define its requirements in a clear, complete, and verifiable manner. During this phase, the problem domain is analyzed thoroughly and both functional and non-functional requirements are specified. This work

package also aims to identify constraints, risks, and external factors that may affect the system design. The analysis results serve as a formal agreement between the developers and users and provide a strong foundation for later design and implementation phases.

Task 4.1 **Problem Analysis and System Modeling:** The Mentora system is analyzed to understand user needs, system boundaries, and interactions between users and system components. Object-oriented analysis techniques are used to model the system and its domain.

Task 4.2 **Functional and Non-Functional Requirements Definition:** Functional requirements describing what the system should do are defined. In addition, non-functional requirements such as usability, reliability, performance, scalability, and supportability are identified and documented.

Task 4.3 **Consideration of Various Factors in Engineering Design:** The impact of public health, safety, welfare, global, cultural, social, environmental, and economic factors on the Mentora system is analyzed. Each factor is discussed individually and its level of effect on the system is evaluated and summarized in a table.

Task 4.4 **Risk Analysis and Alternative Solutions:** Potential technical, ethical, and usability-related risks are identified. For each risk, possible mitigation strategies and alternative solutions (B plans) are discussed and summarized in a risk table.

Task 4.5 **Project Planning and Work Package Definition:** Project goals are reviewed and the overall project is divided into work packages. Start and end dates, responsibilities, milestones, and deliverables are planned. A Gantt chart is prepared to visualize the project schedule.

Task 4.6 Teamwork, Ethics, and Learning Strategy Definition: Plans for ensuring proper teamwork, shared leadership, and equal contribution are documented. Ethical and professional responsibilities are discussed, and learning strategies for acquiring new technical, legal, and interdisciplinary knowledge are defined.

Deliverables
D4.1: Analysis and Requirements Report

WP 5: User Interface (UI) & Backend Design			
Start date: *14.11.2025* **End date:** *17.12.2025*			
Leader	*Sina*	**Members involved:**	*All members*

Objectives: The objective of this work package is to design the user interface and backend structure of the Mentora application in a consistent and user-centered manner. During this phase, the visual layout of the mobile application and the interaction flow between screens are defined. In parallel, the backend architecture is designed to support personalization logic, data storage, and communication between system components. This work package aims to ensure that the UI and backend designs are compatible, scalable, and ready for implementation.

Task 5.1 **User Interface Design and Prototyping:** User interface designs are created to define the layout, navigation flow, and visual components of the Mentora mobile application. Wireframes and mock-ups are prepared to represent key screens such as onboarding, daily study plan, emotion check-in, and progress tracking. ***Task 5.2*** **Backend Architecture Design:** The backend structure is designed using the FastAPI framework, defining API endpoints, data flow, and service responsibilities. The interaction between the frontend and backend is planned using RESTful APIs and JSON-based communication. ***Task 5.3*** **Database Design:** The database schema is designed using PostgreSQL, focusing on storing user profiles, study plans, emotional feedback, and personalization parameters. JSON data structures are planned for flexible storage of dynamic user-related information. ***Task 5.4*** **UI–Backend Integration Planning:** The connection between the React Native frontend and the FastAPI backend is planned, including request–response structures, authentication flow, and error handling mechanisms. This task ensures smooth data exchange and consistent system behavior.
Deliverables ***D5.1:*** *User Interfaces*

4.4 Ensuring Proper Teamwork

Communication and shared responsibility are central to the teamwork throughout the Mentora project. Weekly meetings are held to discuss the progress made, the challenges encountered, and the technical and design-related decisions that need to be addressed. During these meetings, key discussion points are recorded, the date of the next meeting is decided, and the tasks to be completed until then are planned.

We actively contribute to the project by researching topics related to our assigned responsibilities and sharing our findings with the rest of the group. When difficulties arise, we support one another and, when necessary, consult the project supervisor to receive guidance and constructive feedback, ensuring that the project remains aligned with its objectives. Task distribution is handled in a fair and balanced manner by considering individual interests, technical skills, and overall workload.

While we mainly focus on our own areas of interest, we also contribute to other parts of the project when needed, allowing everyone to gain experience with different aspects of the system. Important decisions and task assignments are documented to maintain clarity and continuity, and any differences in opinion are resolved through open discussion and consensus.

4.5 Ethics and Professional Responsibilities

As the Mentora developers, it is important that we obey ethical standards and professional obligations while developing Mentora. We ensure that the features we design are clearly explained to the user and do not create unrealistic expectations since the system interacts with personal information such as emotions, personality traits, and academic habits of users. It should always be clear that Mentora's study plan suggestions are aiming to assist users with their study habits instead of offering psychological diagnoses. We should only collect the data required for the system to operate effectively, and user data must be carefully protected through secure coding techniques and limited access.

Furthermore, the application's message tone should continue to be polite and supportive so that the user doesn't feel overwhelmed by notifications or feedback messages. Moreover, we must obey software engineering standards, record what the system can and cannot do, and be aware of potential threats like privacy issues or unintentional emotional impact while developing the project.

In addition, during the development process, we may make use of artificial intelligence based tools for idea generation, clarification of design decisions, and improvement of proposed solutions. These tools are used as supportive aids rather than authoritative sources. Whenever there is uncertainty or the need for validation, feedback and guidance from course instructors and faculty members are sought to ensure that the final decisions remain academically sound and ethically appropriate.

4.6 Planning for New Knowledge and Learning Strategies

The Mentora project is a multidisciplinary system which combines mobile application development, backend services, data analysis, and user-centered design principles. At the beginning of the project, our team did not have complete prior experience in all of these areas, especially in combining personalized study planning with emotional and personality based analysis. Therefore, acquiring new knowledge and continuously improving technical and domain specific skills is an essential part of the project plan.

From a technical perspective, the project requires learning and applying React Native for cross-platform mobile application development, Python with the FastAPI framework for backend services, and PostgreSQL for structured and semi-structured data storage. Since Mentora relies heavily on JSON-based communication and flexible data representations (such as personality test results and weekly feedback forms), our team plans to study RESTful API design principles, JSON handling strategies, and database design practices that support extensibility. Official documentation, tutorials, and example projects related to React Native, FastAPI, and PostgreSQL will be regularly reviewed, and newly learned concepts will be directly tested within the project implementation.

In addition to core software development, our project includes artificial intelligence based components such as emotion analysis and personalization logic. Although advanced AI models will not be developed from scratch, our team still needs to understand how existing open source models work, what their limitations are, and how their outputs should be interpreted carefully. For this reason, the learning process focuses on the practical use of open source NLP libraries, basic concepts of model evaluation, and integrating AI outputs into the system without overinterpreting the results. Artificial intelligence tools may also be used as supportive learning aids during development, such as exploring alternative design ideas, clarifying implementation details, or generating initial drafts, while all final decisions remain under the responsibility of our team.

Beyond technical knowledge, Mentora requires attention to psychological, legal, and ethical aspects, since the system processes sensitive data such as emotional states, personality traits, and academic habits. To address these concerns, learning strategies are not limited to software engineering topics. The team plans to benefit from consultations with faculty members from psychology and law, as well as experts experienced in innovation and system design. These consultations help the team better understand how personality assessments should be interpreted, how emotional feedback should be presented in a supportive way, and how data protection regulations such as KVKK and GDPR should be reflected in system design. The insights gained from these discussions are incorporated into both functional features and overall system behavior.

Our team has also planned knowledge acquisition as a continuous and collaborative process. Team members research topics related to their assigned responsibilities and share their findings during regular project meetings. This approach helps ensure that learning is distributed across the team and that all members gain a general understanding of the system, even if they focus on different components. When uncertainties arise, the team consults academic resources, expert opinions, and the project supervisor to ensure that decisions are well informed and aligned with project objectives.

Through this ongoing learning strategy, which combines technical self-study, interdisciplinary consultations, team knowledge sharing, and practical experimentation, our team aims to gradually build the required expertise and develop a reliable, ethical, and well-structured personalized study assistant.

5 References

[1]: Object-Oriented Software Engineering, Using UML, Patterns, and Java, 2nd Edition, by Bernd Bruegge and Allen H. Dutoit, Prentice-Hall, 2004, ISBN: 0-13-047110-0.

[2]: Figma Link For Accessing Designs: https://www.figma.com/design/L08CAJBAkyZdcyWfYgX7Z7/Students--Progress-Tracking-App--Community-?node-id=0-1&t=2mjfNgT0NTS6AhgJ-1

[3]: **Assoc. Prof. Dr. Elif Küzeci**

Vice Dean, Faculty of Law, Bilkent University

Expertise: Data Protection Law, GDPR, KVKK (Turkish Personal Data Protection Law), AI Ethics, Legal Philosophy and Sociology, State Theory.

Dr. Küzeci provided guidance on data privacy obligations, explicit consent requirements, processing sensitive data, and legal limitations related to profiling and personal data retention. Her expertise was instrumental in shaping the privacy and ethical approach of the project.

[4]: **Dr. Ezgi Sakman, Ph.D.**

Lecturer, Department of Psychology, Bilkent University

Expertise: Personality assessment, emotional data interpretation, cognitive performance under stress, attachment psychology, individual differences in relationship dynamics, parenting behaviors and child outcomes.

Dr. Sakman advised on the psychological sensitivity of emotional and personality related data, the interpretation limits of such measures, and ethical boundaries when designing supportive study recommendations based on users' psychological profiles.

[5]: **UML Standard**

Object Management Group (OMG). (2017). *Unified Modeling Language (UML) Version 2.5.1*. https://www.omg.org/spec/UML/2.5.1

[6]: **IEEE 830 – Software Requirements Specification**

IEEE Computer Society. (1998). *IEEE Std 830-1998: IEEE Recommended Practice for Software Requirements Specifications*. https://standards.ieee.org/standard/830-1998.html

[7]: **IEEE 12207 – Software Life Cycle Processes**

IEEE Computer Society. (2017). *ISO/IEC/IEEE 12207:2017 Systems and Software Engineering — Software Life Cycle Processes*. https://standards.ieee.org/standard/12207-2017.html

[8]: **IEEE 1016 – Software Design Description**

IEEE Computer Society. (2009). *IEEE Std 1016-2009: IEEE Standard for Information Technology—Systems Design—Software Design Descriptions*. https://standards.ieee.org/standard/1016-2009.html

[9]: **IEEE 1058 – Software Project Management Plans**

IEEE Computer Society. (1998). *IEEE Std 1058-1998: IEEE Standard for Software Project Management Plans*. https://standards.ieee.org/standard/1058-1998.html

[10]: OpenAI. (2025). *ChatGPT*. https://chat.openai.com/