Bilkent University
Department of Computer Engineering

Senior Design Project
T2505
Mind Journal

Analysis and Requirement Report

22002600, Mehmet Eren Anbar, eren.anbar@ug.bilkent.edu.tr
21902557, Deniz Yazıcı, d.yazici@ug.bilkent.edu.tr
22003191, Furkan Mert Aksakal, mert.aksakal@ug.bilkent.edu.tr
22201690, Deniz Şahin, sahindeniz@ug.bilkent.edu.tr
22202218, Ali Deniz Sözer, denizsozer@ug.bilkent.edu.tr
Uğur Güdükbay
Mert Bıçakçı, İlker Burak Kurt

18.12.2025

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

1 Introduction

In today’s busy and stressful world, people are increasingly looking for tools to help them understand their thoughts and handle their emotions. While standard journaling and meditation apps are helpful, they often stay on the surface and don't explain why specific thoughts keep coming back or what triggers them. As life gets more overwhelming, it is hard to figure out these patterns or see how our feelings change over time, which can make us feel disconnected from ourselves.

MindJournal is an AI-powered platform designed to help you explore and organize your thinking patterns. Unlike standard journaling apps that just save your entries, MindJournal uses conversational AI to chat with you naturally, asking relevant questions to help you dig deeper into your ideas. The system automatically spots themes in what you say and groups related thoughts together. It builds a "Mind Map" where you can see your recurring cognitive themes at a glance. It also uses "Thought Chains" to show how one thought triggers another, giving you a clear view of your mental pathways.

Beyond just analyzing your thoughts, MindJournal includes features inspired by Cognitive Behavioral Therapy (CBT). It helps you reframe negative thoughts and offers personalized suggestions for better behaviors. By tracking your progress and what works for you over time, the app supports your long-term growth with practical, actionable guidance.

This report provides an analysis of the project, covering the existing system, the proposed solution, and its requirements. Key aspects such as system models, functional and non-functional requirements, engineering considerations, risks, and project planning are discussed. The report also highlights teamwork strategies, ethical considerations, and approaches for addressing potential challenges.

2 Current System

The mental wellness and self-reflection market is currently populated by a wide variety of journaling applications, AI-based companions, and CBT-oriented therapeutic tools. While these solutions offer value through emotional support, guided journaling, or structured well-being exercises, they generally operate as fragmented tools rather than unified systems. Consequently, the current market lacks a single integrated platform capable of connecting organic thoughts, behavioral patterns, and long-term cognitive development.

In the domain of AI-based emotional support, applications like Wysa [1] utilize chatbots to provide empathetic conversation and basic CBT tools. Although effective for short-term emotional regulation, Wysa treats interactions primarily as isolated sessions. It fails to automatically cluster user thoughts, analyze recurring themes, or maintain long-term cognitive patterns across different conversations, limiting its ability to provide structured insight.

Guided journaling platforms also present specific limitations regarding deep cognitive analysis. Mindsera [2], offers AI-guided journaling with cognitive reframing but organizes entries in a linear structure, failing to form interconnected thought categories or track behavioral history. Similarly, Reflectly [3] and Clearful [4] focus heavily on aesthetic prompts and mood questionnaires. While visually engaging, these applications lack deeper psychological analysis and do not offer capabilities for clustering thoughts or tracking the effectiveness of behavioral interventions. Stoic [5], while refined for daily reflection and philosophy, also fails to reveal hidden cognitive patterns or visualize the "thought chains" that connect different mental states.

Furthermore, structured therapeutic and mindfulness applications often rely on pre-set content rather than user-generated exploration. Platforms such as MindDoc [6] and Bloom [7] function as digital mental health programs offering predefined modules and assessments. They do not support a conversation-based journaling interface or the automatic detection of recurring thought patterns, as they are designed around rigid lessons rather than organic interaction. Leading meditation apps like Calm [8] and Headspace [9] specialize in stress regulation through external content like sleep stories and guided practice. However, they lack tools for internal cognitive processing, thought structure analysis, or behavioral change tracking.

3 Proposed System

3.1 Overview

MindJournal is a comprehensive AI-assisted thought analysis platform designed to help users understand their thoughts and manage mental patterns by offering deep insights into recurring cognitive themes and underlying triggers. Unlike traditional journaling and meditation applications, which often fail to provide structured feedback or identify long-term behavioral patterns, MindJournal introduces a data-driven approach to explore, organize, and visualize the user's psyche. This innovation addresses the difficulties users face in identifying why certain thoughts repeat and how their emotional states evolve, preventing a fragmented understanding of their internal experiences.

The platform integrates Natural Language Processing (NLP) and advanced cognitive analysis to deliver detailed psychological profiles. Conversational AI is utilized to collect thoughts through natural interactions and prompt reflective questioning, while dynamic categorization algorithms group related thoughts into a "Mind Map" to visualize cognitive themes. By combining "Thought Chain" analysis to track triggers and CBT-inspired reframing for behavioral intervention, MindJournal offers a comprehensive system for self-reflection that surpasses the capabilities of passive recording tools.

MindJournal is developed using a three-layered architecture, utilizing React Native for the mobile application and Node.js for the backend web server, ensuring a responsive user experience and cross-platform compatibility. For data management, an SQL storage unit is employed to handle both raw journal entries and processed analytics, while a scheduled "Thought Classifier LLM" processes batch data to maintain performance. The platform's modular design, incorporating lightweight on-device models and scalable backend services, allows for efficient processing and seamless integration of future cognitive analysis features.

3.2 Functional Requirements

3.2.1. User Account & Personal Profile Management

- The user must be able to create an account using an email and password combination.
- The user must be able to securely log in and log out of the application at any time.
- The user must be able to reset their password via email based verification.
- The user must be able to update personal settings.

- The user must be able to permanently delete their account, which must remove all associated thoughts, categories, behaviors, and progress history.
- All user data must be stored separately so that each user interacts only with their own journaling content, categories, and statistics.

3.2.2. Thought Journaling

- The user must be able to start a new journaling session from the “Start Journal” interface.
- At the beginning of each session, a chatbot must greet the user with an initial prompt such as “What’s on your mind right now?”.
- The user must be able to submit thoughts by recording audio or typing.
- After the user shares a thought, the chatbot must respond with context-aware, reflective, and open-ended guiding questions to help the user explore the thought further.
- The chatbot must additionally ask whether the user has any other thoughts they would like to share, enabling the journaling session to collect multiple thoughts within the same conversation flow.
- Every thought expressed by the user during the session must be automatically analyzed and associated with the most relevant category, ensuring that recurring or thematically similar thoughts are grouped under the same category node.
- The user must be able to end the journaling session at any time by selecting the provided exit option.
- The user must be able to review previously completed journaling sessions, including all thoughts captured during the conversation.

3.2.3. Mind Map Visualization of Thought Categories

- The user must be able to access the “Mind Map” screen from the main menu.
- The mind map must display categories as nodes, visually representing groups of similar or recurring thoughts.
- The user must be able to zoom in, zoom out, and drag across the graph to explore different regions of the map.

- The user must be able to view the list of thoughts associated with any selected category node.
- The size and visual prominence of each node must directly reflect the number of thoughts associated with that category.

3.2.4. Thought Statistics

- The user must be able to select any category to view detailed statistical information.
- The user must be able to view the recurrence frequency of thoughts within that category.
- The user must be able to filter statistics by daily, weekly, monthly, or custom date ranges.
- The user must be able to view the emotional patterns associated with the thoughts in that category.

3.2.5. Thought Chain & Trigger Relationship Analysis

- The user must be able to access a “Thought Chain” view that visualizes how thoughts influence or trigger one another.
- The user must be able to see directed edges (arrows) showing relationships such as cause, trigger, or reinforcement.
- The user must be able to select any connection to view the initiating thought and the emotional state reflected in that chain.
- The user must be able to click the “Break the Chain” button to request cognitive reframing assistance.

3.2.6. CBT Reframing

- The user must be able to select a specific negative or distorted thought to request a CBT-based reframing.
- The user must receive a reconstructed, rational, and balanced alternative interpretation of the thought.
- The user must be able to compare their original thought with the reframed version.

3.2.7. Behavioral Intervention Selection & Feedback

- The user must be offered behavior suggestions that align with the thought after receiving a CBT reframing.
- The user must be able to either select an AI-suggested behavior, enter a custom behavior of their own, or skip the step entirely.
- The user must be able to associate the selected behavior with both the specific thought and its corresponding category.
- The user must be able to provide effective feedback for any chosen behavior using “It worked”, “It didn’t work”, or “Remind me later.”
- When a new thought is added to a category that contains previously recorded behaviors, the user must be presented with previously successful behaviors as suggestions.
- The user must receive reminders when the system detects a recurring thought pattern that previously required behavioral intervention.
- The user must be able to view a complete history of selected behaviors, their associated thoughts/categories, and success rates in the “Behavior Progress” section.

3.2.8. Improvement Tracking & Psychological Trend Monitoring

- The user must be able to select a category to view long-term improvement trends.
- The user must be able to view both the reduction in thought frequency and the behavioral success percentages.
- The user must receive positive reinforcement notifications when improvement is detected (e.g., “Great job! This category shows a 25% reduction this month.”).
- The user must receive early warning notifications when a category becomes more active again.
- The user must be able to see weekly or monthly summaries of mental progress.

3.2.9 Onboarding QA

- The user must be able to answer onboarding questions asked to them.
- The age, gender, preferences and any other related attribute that would contribute to the thought classification process for a user must be obtained.

3.3 Non-functional Requirements

3.3.1 Usability

- The system shall provide an easy to use interface for users to journal their thoughts, navigate through different segments and follow categories in the mindmap.
- All the occurring functionality must be easy to use, find and learn. The user shall not struggle to understand any feature.
- When logged in for the first time, users shall be welcomed by a tutorial to the app, minimizing any possible reasons for question marks.
- MindJournal shall be compatible both for IOS and Android devices without any component losses or disintegrations.

3.3.2 Reliability

- The system must deliver accurate analysis for input thoughts and connect them to related categories.
- There must never be any treatment statements in the given report to the user due to legal reasons.
- Comments given by the LLM must strictly follow the required structure that is retrieved either from datasets or training strategies.
- The app should be able to keep on working when faced with any errors or problems.

3.3.3 Performance

- Our chatbot will take 3 seconds to respond based on context to the given thought input.
- Mind map creation and the chains between the thoughts will take a longer time as there will be a large number of parameters to check while generating. It will still take less than a minute.

3.3.4 Supportability

- The system should be object oriented and follow best practices of software development.
- DRY and SRP principles should be followed in order to prevent any redundancies.
- The code should be easy to follow and understand for the existing functionalities.

- The codespace should be well documented for others to act in accordance with the implementation logic.

3.3.5 Scalability

- The bottleneck of the proposed system under heavy load is LLM inference. Therefore, for scalability purposes, we propose a batched and scheduled LLM processing on journaling sessions. Moreover, selecting node as web server layer to enable multi-threaded processing of requests.

3.4 Pseudo Requirements

This section specifies development-related requirements that are not directly tied to system functionality but are essential for successful implementation, validation, and maintenance of the MindJournal system.

3.4.1 Version Control

- GitHub shall be used as the primary version control system.
- A shared GitHub repository shall be maintained for all project artifacts.
- Pull requests shall be reviewed by at least one team member before merging.

3.4.2 Technology Stack and Tools

- **Mobile Application:** React Native
- **Backend:** Node.js
- **Database:** SQL based Relational DB
- **LLM Components:** Lightweight LLMs for journaling and CBT
- **Documentation:** Markdown and PDF

3.4.3 Testing and Validation

- For core backend services like authentication, journaling, categorization and CBT modules unit testing shall be executed.
- The communication between mobile client, backend services and database shall be verified via integration testing.
- Functional validation shall be managed by verifying system behaviour against the use cases.
- Error handling is used to test the recovery from invalid inputs or other failures.

3.5 System Models

This section presents behavioral and structural models of the MindJournal system using UML-based representations.

3.5.1 Scenarios

Use Case: User Registration and Login

Actors: User
Entry Condition: User opens the application
Exit Conditions:

- User successfully logs in
- Authentication fails

Flow of Events:

1. The user enters an email and password.
2. System validates credentials.
3. The system creates a session and grants access.

Use Case: Start Journaling Session

Actors: User
Entry Condition: User is logged in
Exit Conditions:

- Session is saved
- User exits journaling

Flow of Events:

1. The user initiates a new journaling session.
2. Chatbot asks an initial reflective question.
3. The user submits thoughts via text or audio.
4. Chatbot generates context-aware follow-up questions.
5. Session is stored upon user exit.

Use Case: Thought Categorization

Actors: System
Entry Condition: Journaling session ends
Exit Conditions:

- Thoughts categorized successfully

Flow of Events:

1. The system extracts thoughts from the session.
2. Rapid category checker maps thoughts to existing categories.
3. If needed, Thought Classifier LLM creates or refines categories.
4. Categorized data is stored.

Use Case: View Mind Map

Actors: User
Entry Condition: User accesses Mind Map screen
Exit Conditions:

- User exits visualization

Flow of Events:

1. The system retrieves thought categories and statistics.
2. Mind map nodes are rendered based on frequency.
3. The user explores categories and associated thoughts.

Use Case: CBT Reframing

Actors: User, System
Entry Condition: User selects a negative thought
Exit Conditions:

- Reframed thought displayed

Flow of Events:

1. The user requests CBT assistance.
2. CBT Service generates a balanced reinterpretation.
3. The system displays original and reframed thoughts.

3.5.2 Use-Case Model

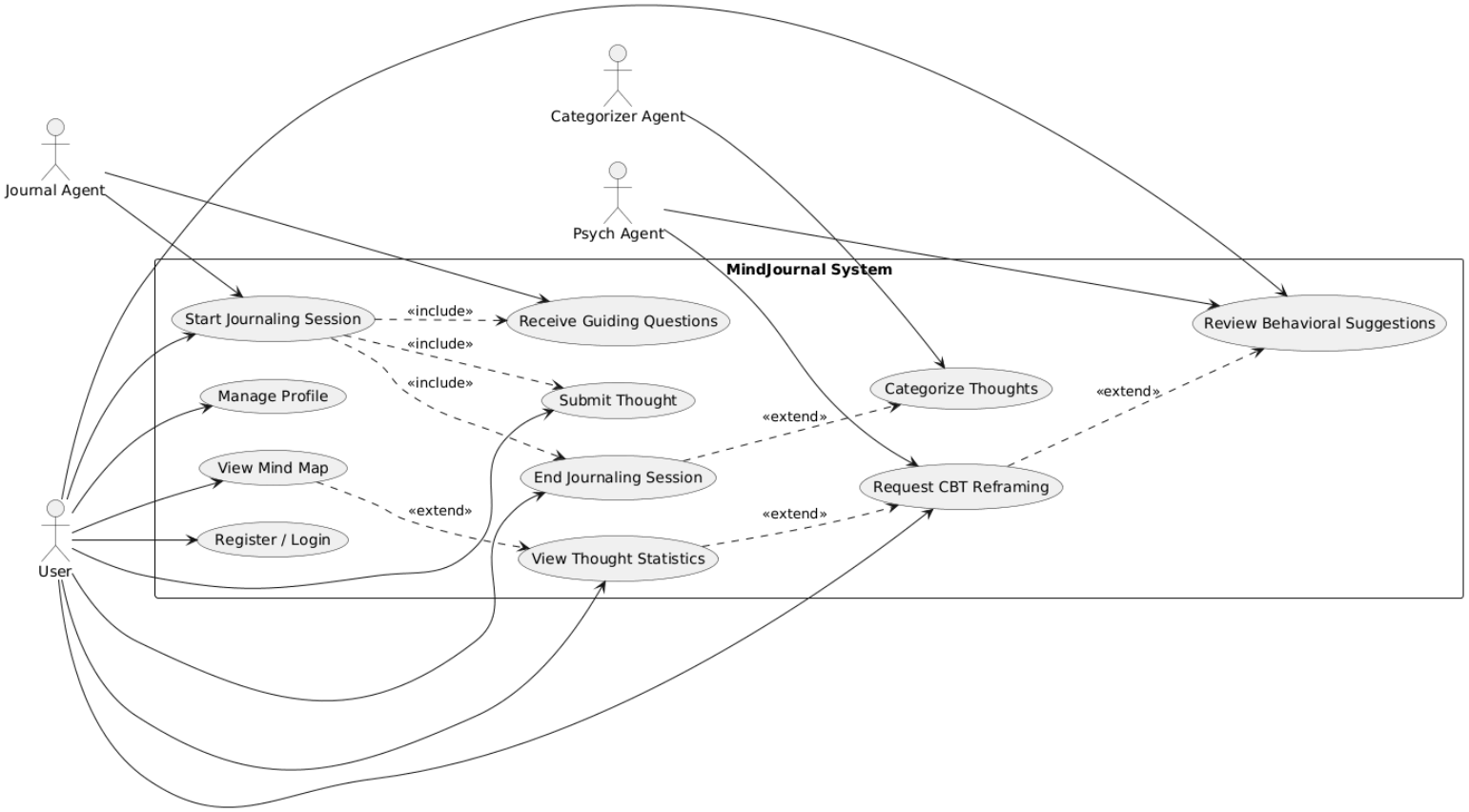

Fig 1. Use Case Model

3.5.3 Object and Class Model

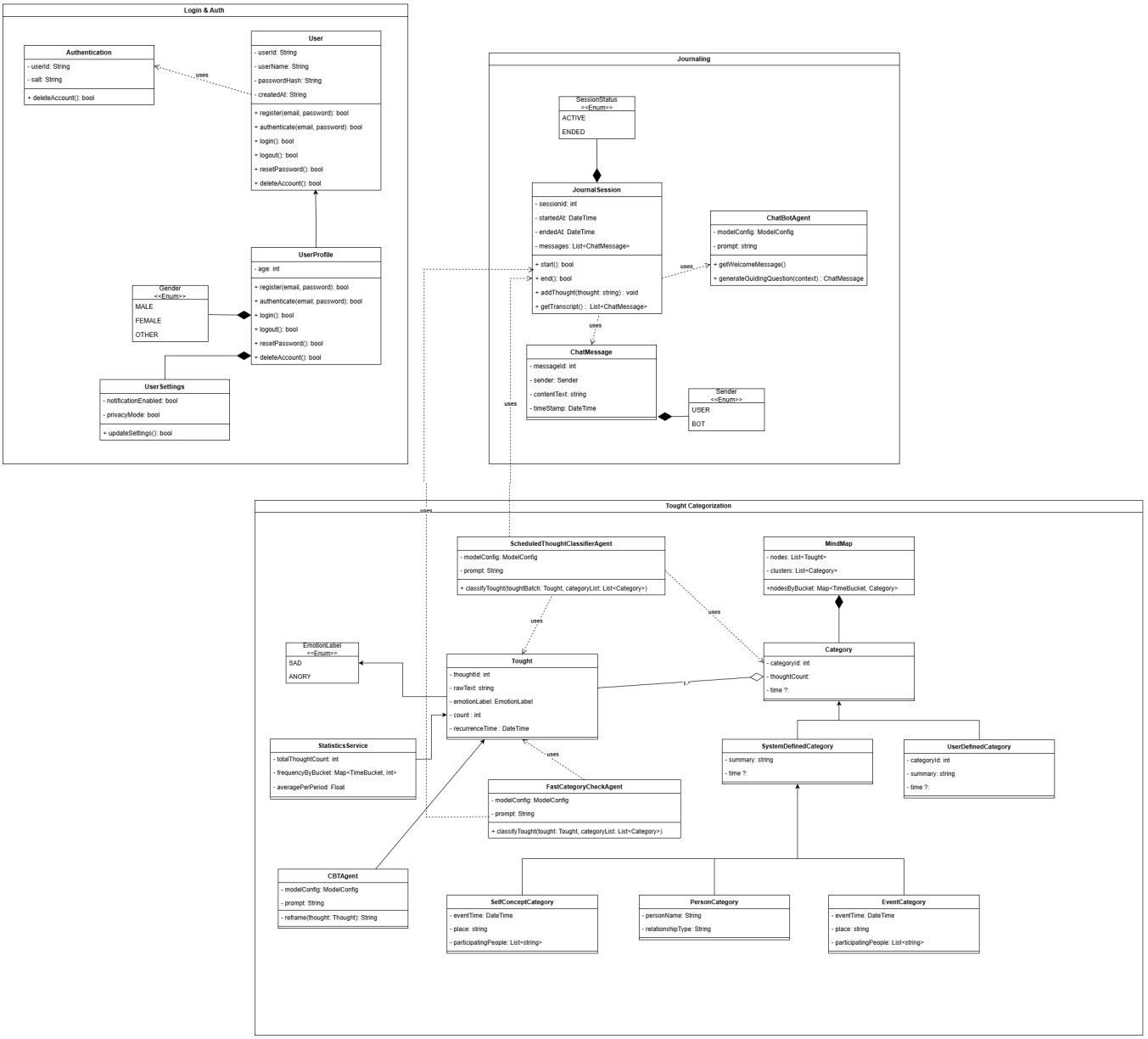

Fig 2. Object and Class Model

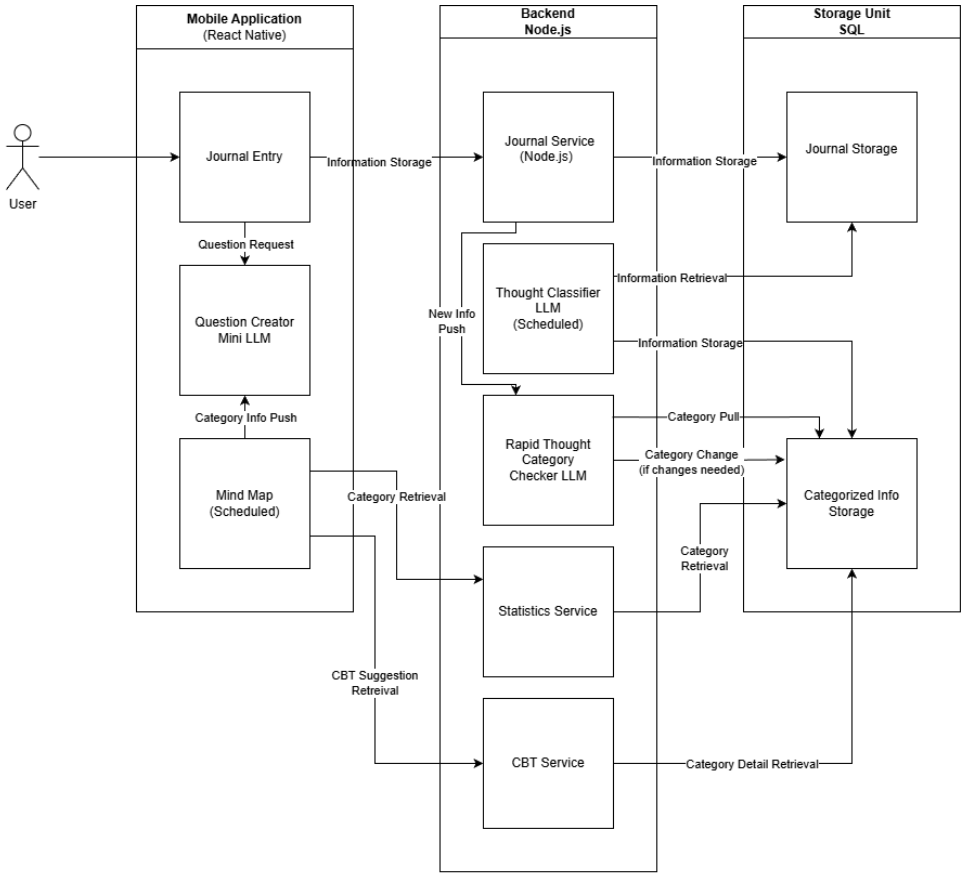

Fig 3. Class and Object Model

User: Represents a MindJournal account. Stores authentication identity

UserProfile: Stores non-auth profile information and settings such as user-specific preferences and onboarding attributes that may affect classification. Owns all journal data and analytics. (notification preferences, language, onboarding answers).

Authentication: Handles user registration, login/logout, password reset, and account deletion requests.

User settings: Represents settings specified by the user.

JournalSession: A single journaling interaction instance. Stores timestamps, session state, and the list of question answer pairs captured during that session.

ChatMessage: Represents an individual message exchanged during a journaling session, either from the user or an agent. Stores message content, sender role, and timestamp.

ChatBotAgent: Acts as the primary conversational interface between the user and the system. It coordinates dialogue flow, routes user input to relevant agents, and delivers system-generated responses during journaling sessions.

ScheduledThoughtClassifierAgent: Periodically analyzes stored thoughts in the background to refine or update their categorical assignments. This agent supports delayed or improved classification without interrupting active user interactions.

MindMap: Represents a structured visualization of the user's thoughts and their relationships to categories over time. It aggregates categorized thoughts to reveal recurring themes, connections, and self-concepts.

Category: Abstract representation of a conceptual label used to group thoughts. Categories provide semantic structure for organizing, analyzing, and visualizing journaling data.

SystemDefinedCategory: Represents predefined categories supplied by the system to ensure consistent baseline classification. These categories cannot be modified or deleted by users.

SelfConceptCategory: Represents categories related to the user's beliefs, traits, or identity-related statements. These categories are used to track recurring self-perceptions and cognitive patterns.

PersonCategory: Represents categories associated with specific people mentioned in thoughts. Enables aggregation of thoughts centered around interpersonal relationships.

EventCategory: Represents categories linked to specific events or situations described in thoughts. Useful for identifying event-driven emotional or cognitive patterns.

UserDefinedCategory: Represents custom categories created and managed by the user. These allow personalization beyond system-defined classifications.

Thought: Represents a single unit of user-generated cognitive content captured during journaling. A thought may be linked to one or more categories and serves as the core data element for analysis and visualization.

FastCategoryCheckAgent: Performs lightweight, real-time category estimation for newly submitted thoughts. Its goal is to provide immediate feedback while deferring more complex analysis to background agents.

StatisticsService: Aggregates and computes analytics over thoughts, categories, sessions, and time. Provides quantitative insights such as frequency trends, category distributions, and behavioral progress indicators.

CBTAgent: Supports cognitive-behavioral reflection by identifying cognitive patterns and generating structured prompts or reframing suggestions. It operates as a supportive analytical agent without performing clinical diagnosis.

Thought: The main content unit extracted from user inputs. A thought can be linked to a category, carry emotional signals, participate in thought chains, and be the target of CBT reframing.

3.5.4 Dynamic Models

Some diagrams for the base functionality of our project are given below.

3.5.4.1 Share Thoughts Sequence Diagram

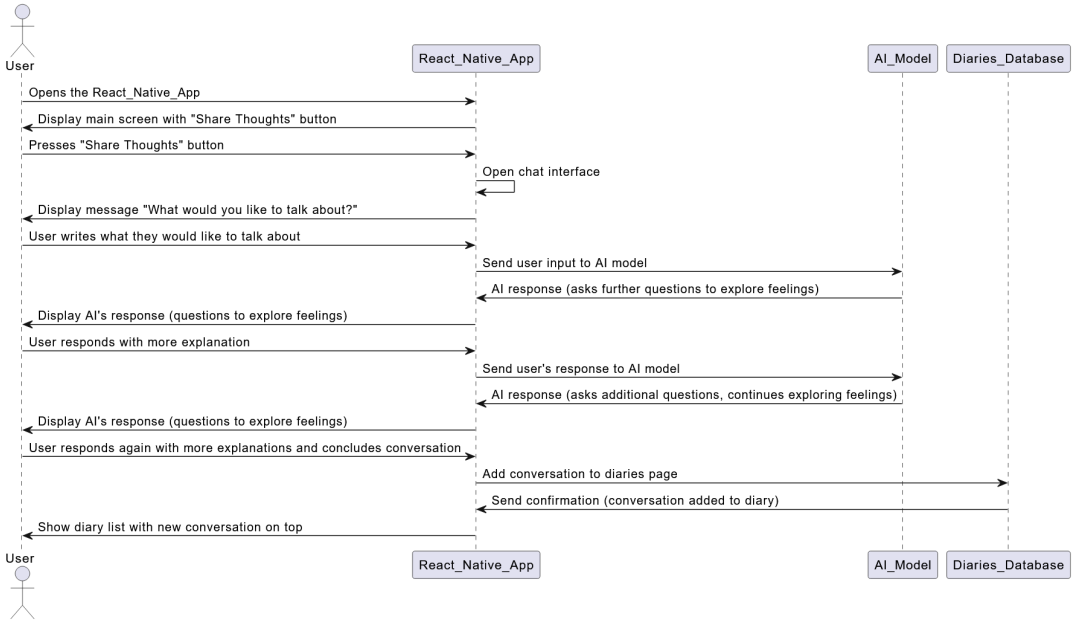

Fig 4. Share Thoughts Sequence Diagram

3.5.4.2 Authentication Activity Diagram

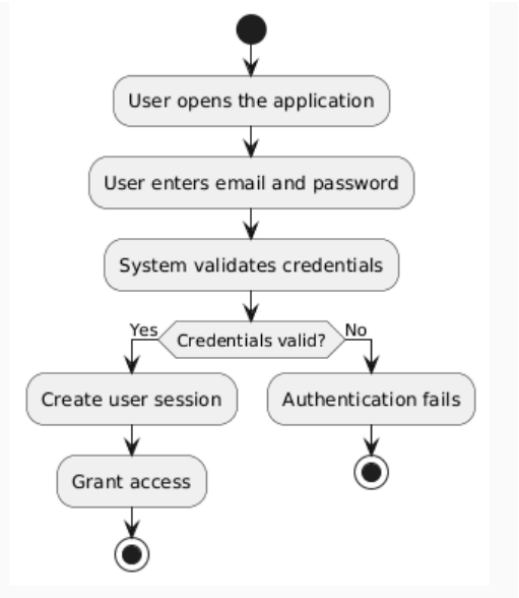

Fig 5. Authentication Activity Diagram

3.5.4.3 Thought Categorization

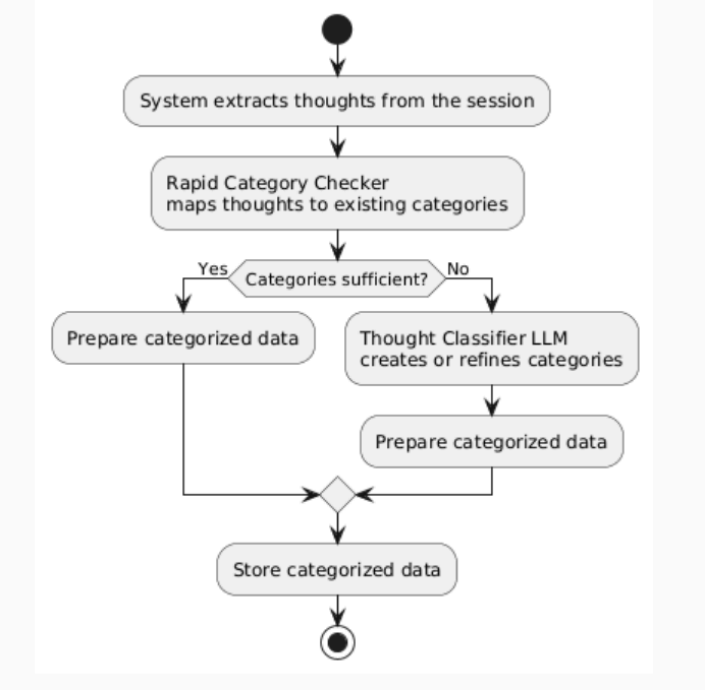

Fig 6. Thought Categorization Activity Diagram

3.5.4.4 Request Thought Map Sequence Diagram

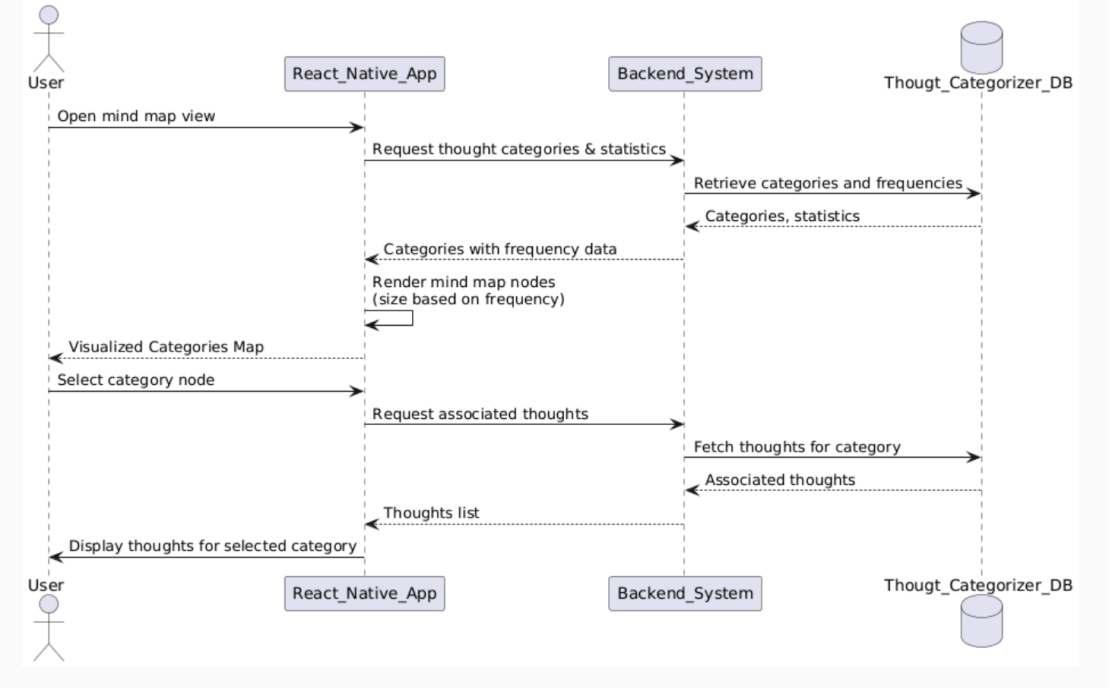

Fig 7. Request Thought Map Sequence Diagram

3.5.4.5 CBT Request Sequence Diagram

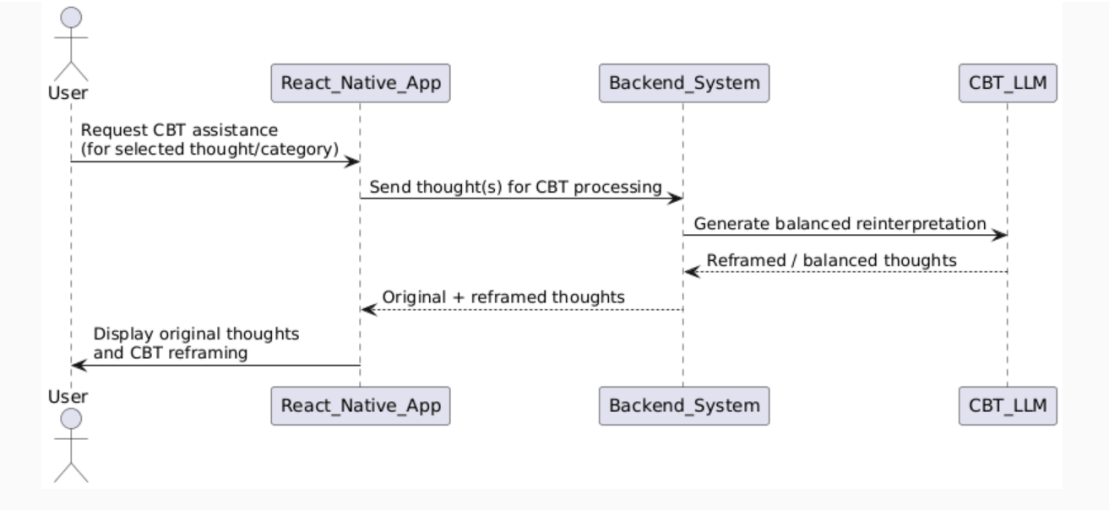

Fig 8. CBT Request Sequence Diagram

3.5.4.6 Category Statistics Request Sequence Diagram

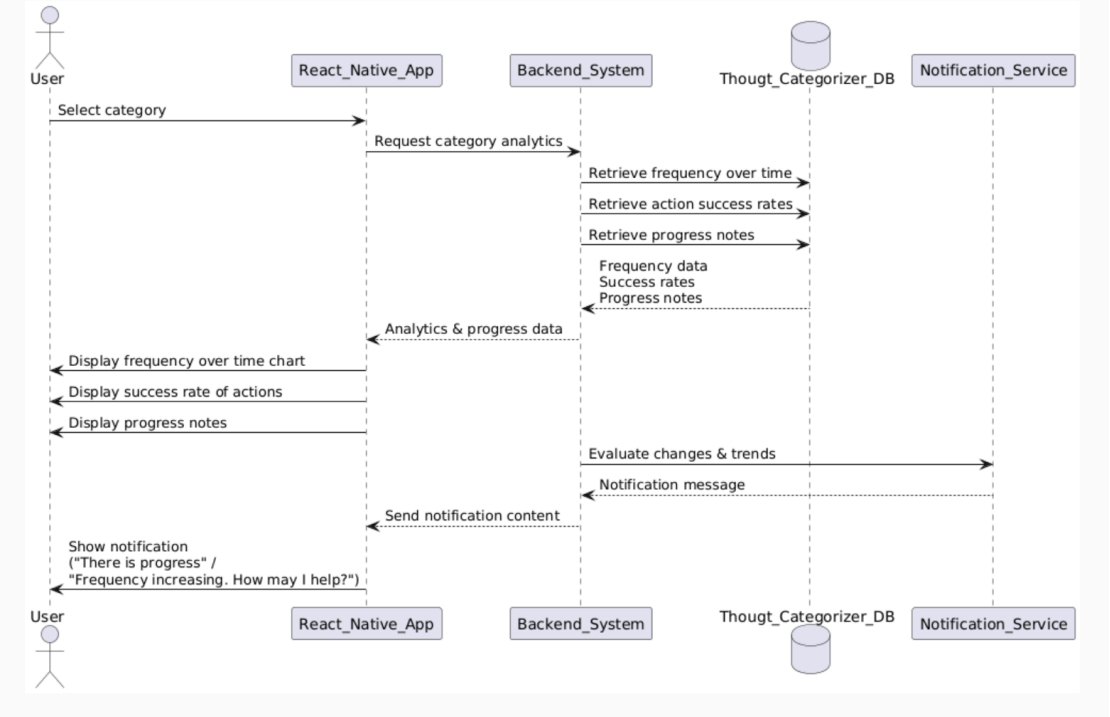

Fig 9. Category Statistics Request Sequence Diagram

3.5.5 User Interface

3.5.5.1 Login Page

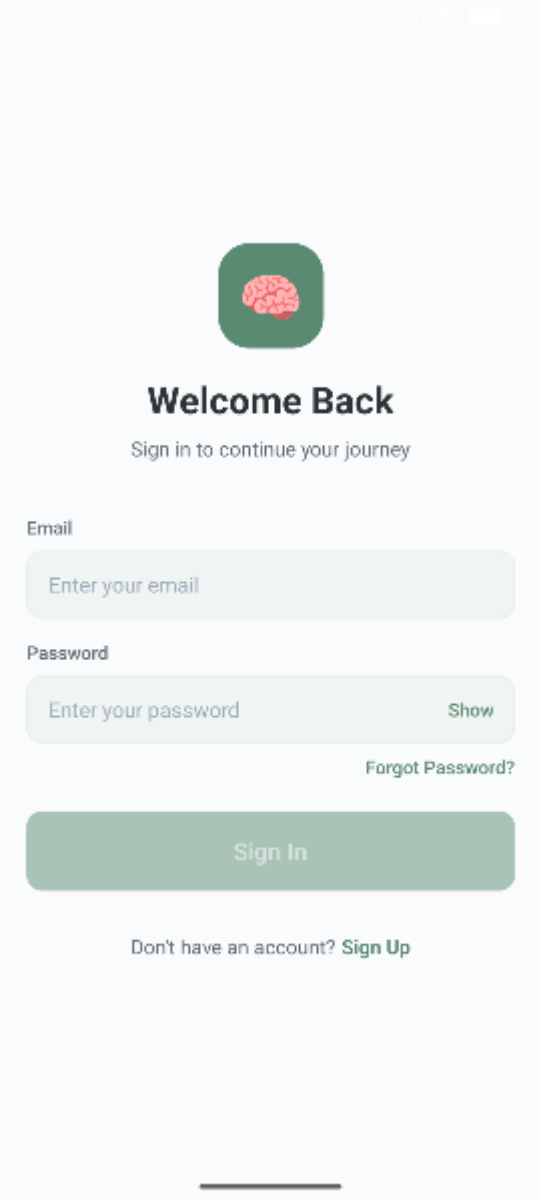

Fig 10. Login Mockup

This is the screen that the users who are opening the first time or have logged out will see. The user needs to enter their e-mail and password to enter the app. If the given login information is correct the user is redirected to the landing page.

3.5.5.2 Register Page

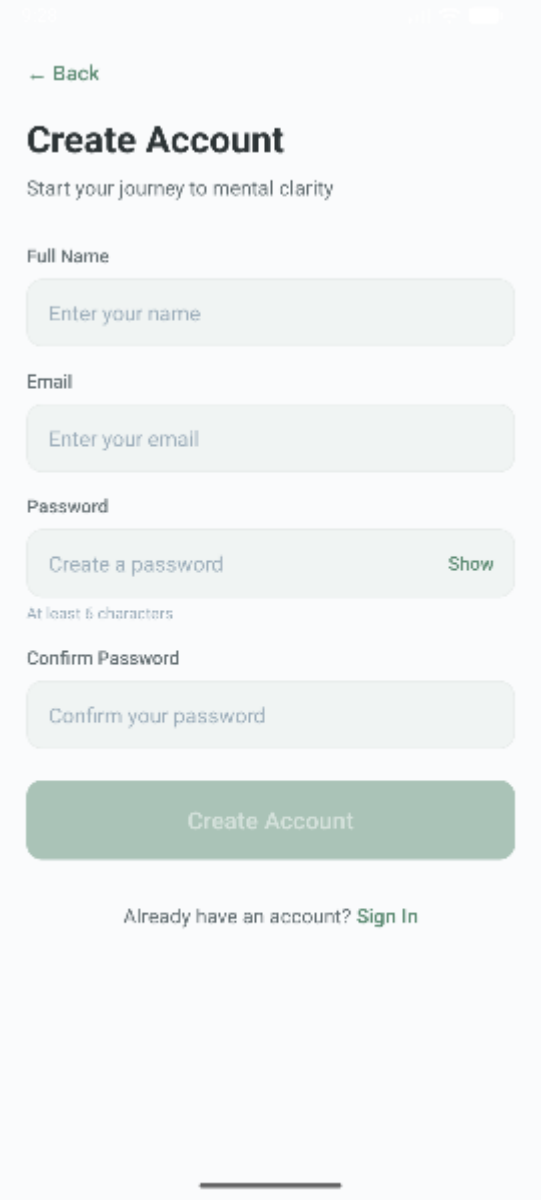

Fig 11. Register Page Mockup

This is the page a user enters when they click sign up on the entry screen or when they open the app for the first time. The user can enter their name, e-mail and password in order to create a new account. Once an account is created the user is redirected to the landing page.

3.5.5.3 Journal Page

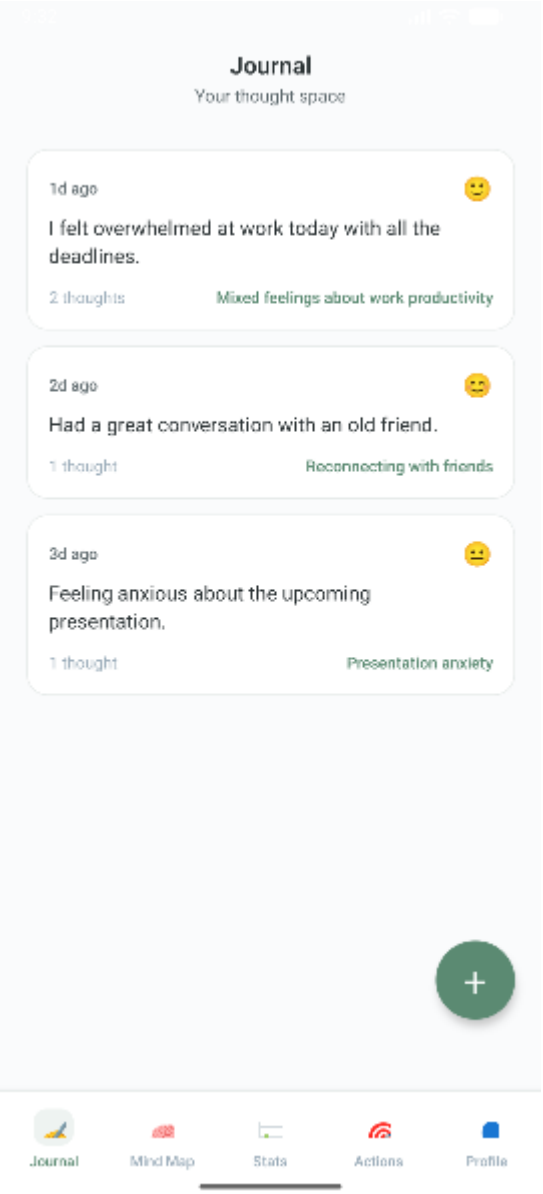

Fig 12. Journal Page Mockup

The journal page will be used as the page where the user can see his past journal entries and start a new journal entry. This page will contain unsorted journal entries just by giving them in entry order.

3.5.5.4 Onboarding Page

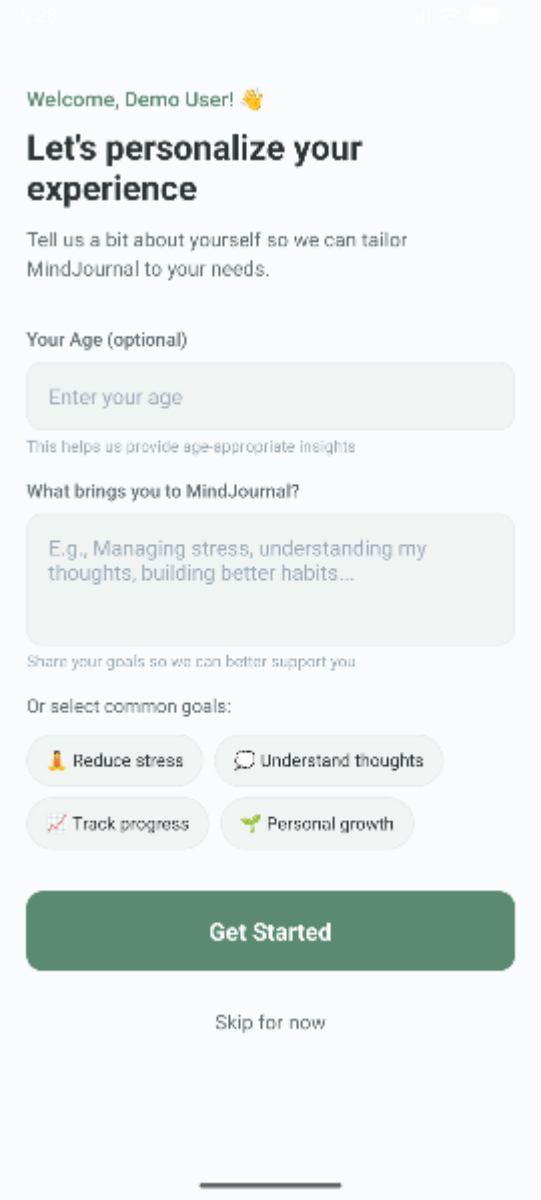

Fig 13. Onboarding Page Mockup

This page allows the user to enter personal information and their goals. The input from the user is used later on to customize their experience. This is an optional page and either button leads to the home landing page.

3.5.5.5 Conversation Page

Fig 14. Conversation Page Mockup

This page allows the user to have a conversation with the LLM agent. The user can end this conversation anytime they want and will be directed to the journal page when the conversation is over.

3.5.5.6 Session Overview Page

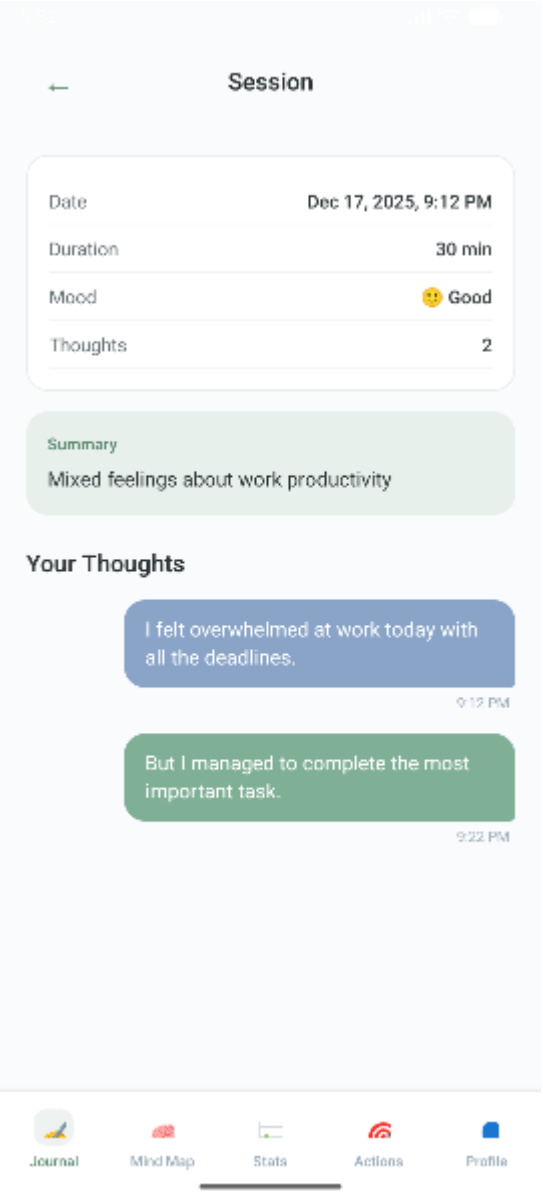

Fig 15. Session Overview

This page will contain the overview of a journaling session and will be accessible by clicking on a session through any interface where they are visible (journal page, stats, etc.) This page will show the messages sent in the conversation, the categorization of said conversation and all the other AI generated analysis specifically of this session.

3.5.5.7 Mind Map Page

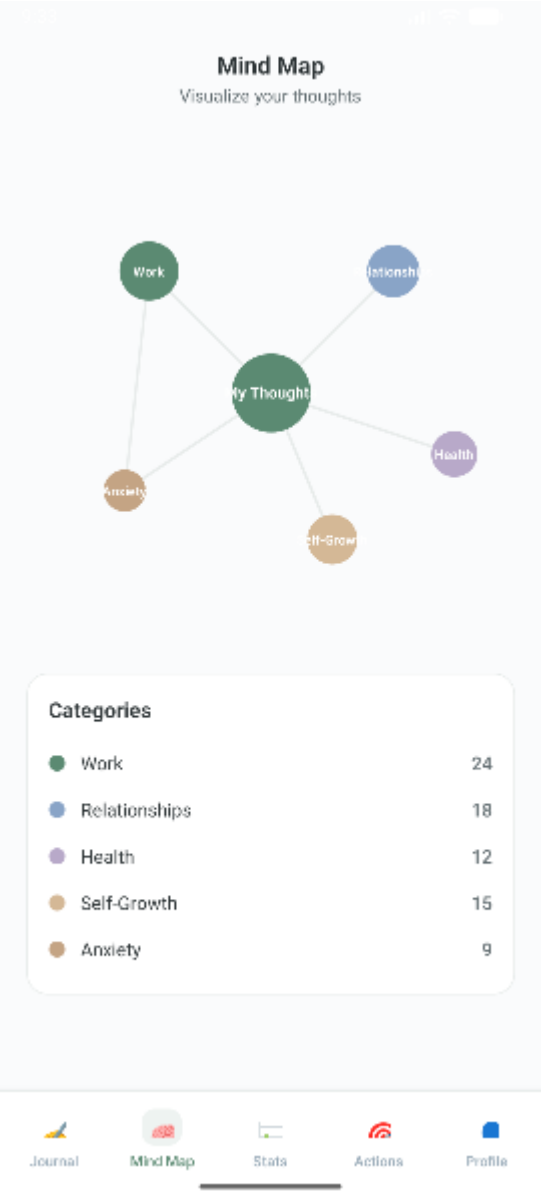

Fig 16. Mind Map Page

This page visualizes the results from the user's journal into a graph form. The user can see their thoughts split into categories and see which categories are being talked about more frequently with the LLM agent. This can be either done by looking at the number statistics or looking at the size of the nodes in the mind map.

3.5.5.8 Insights Page

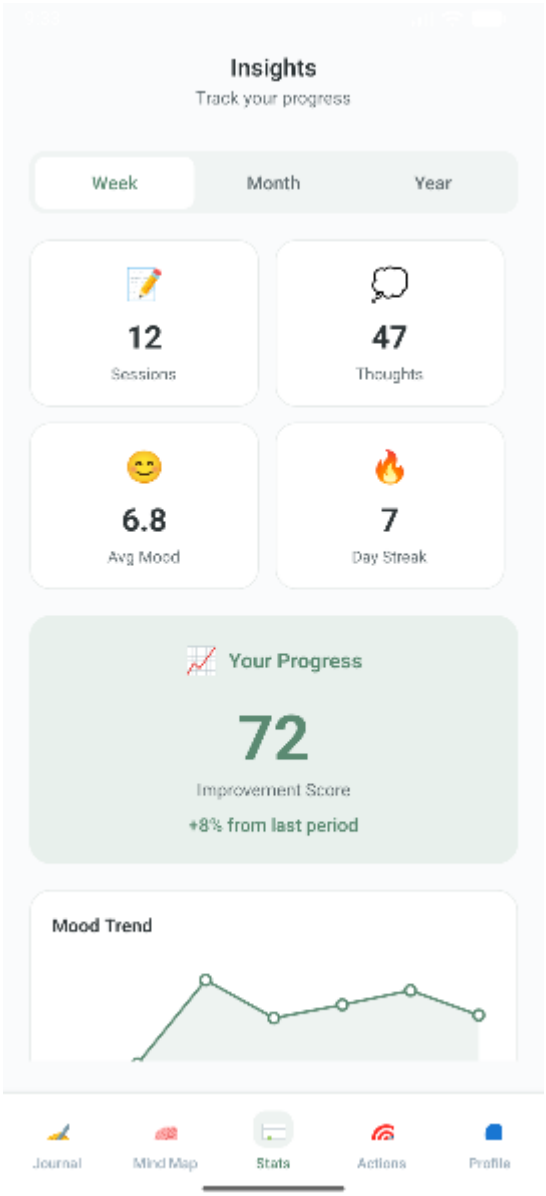

Fig 17. Insights Page

The insights page contains information on user progress. Information such as journal entry count, how the user mood is moving and so on is visualised for the user here. This page contains information for the user to see the overall progress without the specifics of categorization and individual thoughts. The insights are also viewable weekly, monthly, yearly for further understanding of one's own psychology.

3.5.5.9 Actions Page

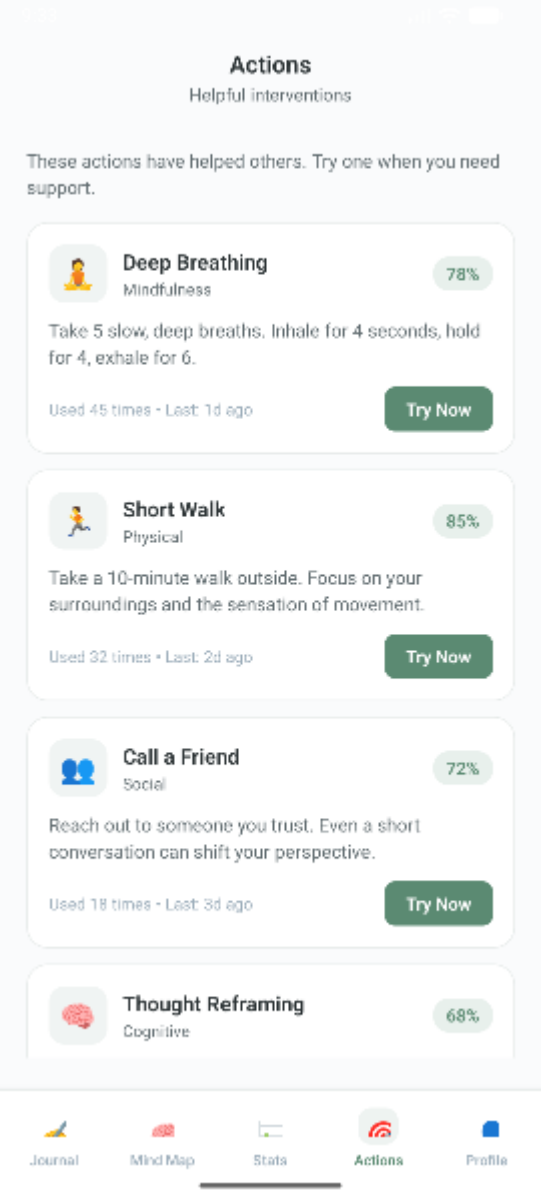

Fig 18. Actions Page

This page shows the user a set of actions that the user can do to calm down. The percentages in the top right corner of the actions represent how many of the other users found that action helpful. Pressing “Try Now” opens the review popup.

3.5.5.10 Action Review Popup

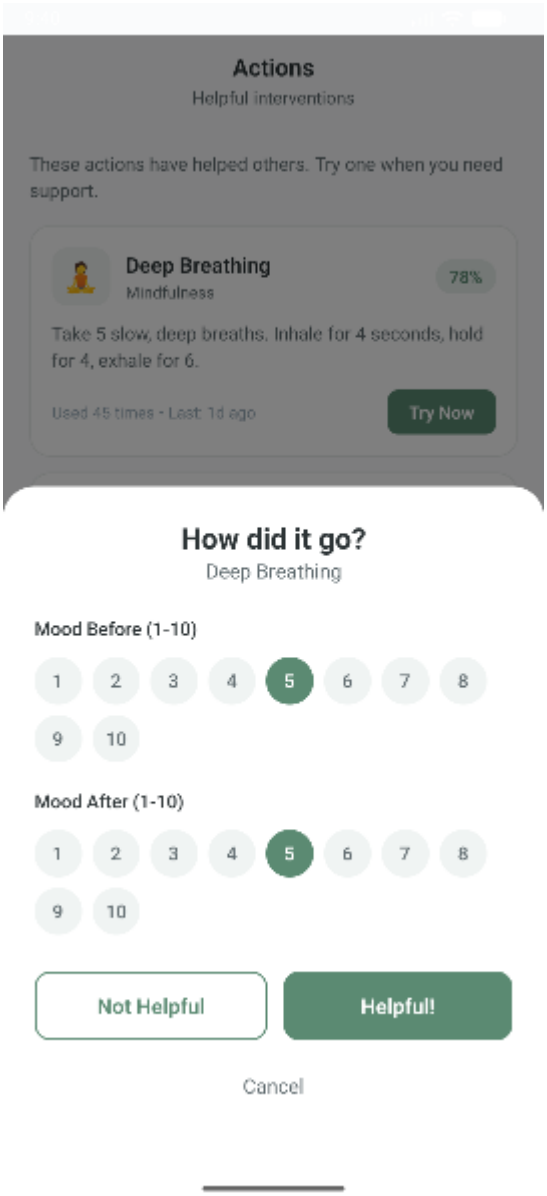

Fig 19. Action Review Popup

This popup appears after a user selects an action from the actions page. The user is able to rate their experience of the exercise. The data collected by this entry then affects how often actions are advised to users and are displayed on the actions page.

3.5.5.11 Profile Page

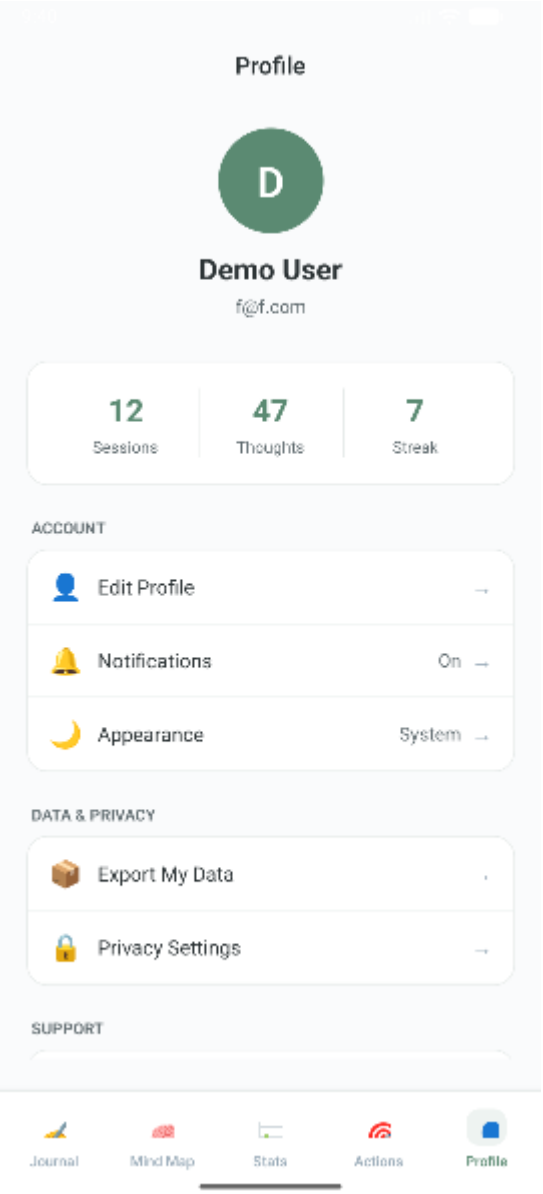

Fig 20. Profile Page

This page contains profile information and general settings the user can change. From this page the user can edit their profile, switch between light and dark mode. Also they can access privacy related settings. This page is also where the user can logout from.

3.5.5.12 Landing Page

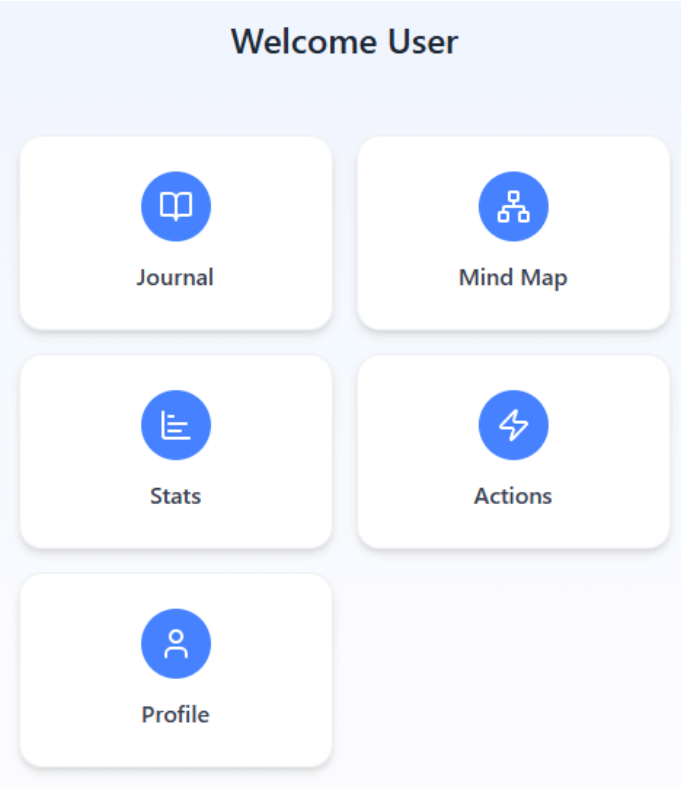

Fig 21. Landing Page Mockup

This page is the first page the user will see after they successfully login. The user can choose which page they want to go from here.

4 Other Analysis Elements

4.1 Consideration of Various Factors in Engineering Design

This section presents the major engineering factors influencing the design and implementation of the MindJournal system. Because the application will process sensitive personal data and integrate AI-driven cognitive analysis, technical considerations along with ethical, social, and environmental ones should be carefully developed to make it a responsible, sustainable, and high-quality system.

4.1.1 Constraints

4.1.1.1 Implementation Constraints

MindJournal is designed as a mobile-first system supported by backend services and AI-driven processing components, imposing several implementation-related constraints. It needs to work seamlessly on iOS and Android, which means it has to support cross-platform approaches to development, limiting platform-specific optimizations. Moreover, a great part of the system's functionality relies on natural language processing and large language model inference, which sets constraints related to response time, model size, and inference efficiency. The road map for future AI components on the mobile side includes guiding question generation and light interaction logic; thus, the complexity of the models is strongly limited by mobile hardware characteristics, such as memory, processing power, and battery consumption. On the server side, journaling sessions need to be processed in a modular and scalable way, calling for a service-oriented and object-oriented architecture with support for scheduled and batched inference. Secure and reliable communication among the mobile application, the backend services, and the storage layers is mandatory, as journaling data is constantly generated and processed during the lifecycle of the system.

4.1.1.2 Economic Constraints

The economics of MindJournal dominate both the architecture and the deployment strategy of the system. A subscription-based business model requires careful control over infrastructure costs and operational expenditures, particularly in early deployment phases. In this respect, the development relies as much as possible on open-source libraries, public APIs, and managed cloud services to reduce development expenses. Nevertheless, long-term journaling is likely to cause steady growth in storage requirements for raw journal entries, categorized thoughts, behavioral histories, and statistical summaries, which may raise operational costs. Moreover,

frequent invocation of large language models for the purpose of thought analysis, CBT reframing, and behavioral recommendations introduces periodic computation costs; optimization strategies such as batching, caching, and selective processing will become economically necessary.

4.1.1.3 Ethical Constraints

MindJournal deals with highly sensitive personal and psychological data, which adds powerful ethical constraints on system design and behavior. What users think, how they feel, and their behavioral feedback has to be treated as strictly confidential information and protected from unauthorized access or misuse. The system cannot present itself as a medical or therapeutic authority and must not diagnose, treat, or make claims that could be understood as professional psychological advice. All AI-generated outputs, including CBT-inspired reframes and behavioral suggestions, shall remain supportive, neutral, and non-judgmental in tone. Users will need to provide explicit consent around data gathering and processing. Users will be informed in detail about where, how, and for what purpose their data is stored, analyzed, and used. In addition, it provides assurance that users fully control their data, including the right to delete their account and all associated records permanently.

4.1.1.4 Social Constraints

Social factors strongly affect the usability and acceptance of the MindJournal system. The application needs to be available for users at different stages of technical proficiency and psychological self-awareness, thus requiring intuitive interaction flows and clear explanations of the features offered by the system. AI components need to use culturally sensitive language and must be carefully designed in such a way that user experiences are neither invalidated nor stigmatized. Since the system interacts with users' emotional and cognitive patterns, it needs to be designed to promote reflection rather than dependency, where users do not regard the AI as an alternative to human support. The visual components, such as mind maps, chains of thoughts, and statistical summaries, need to balance informativeness against clarity, avoiding overwhelming or confusing representations which may have adverse implications for user engagement.

4.1.2 Standards

IEEE 830: IEEE 830 provides a recommended practice for Software Requirements Specifications (SRS), emphasizing correctness, completeness, and consistency. By following this standard, the project minimizes ambiguity and establishes a structured baseline for validation by clearly documenting both functional and non-functional requirements.

ISO/IEC25010: ISO/IEC 25010 defines a product quality model to evaluate software systems. The project utilizes this framework to assess nine critical characteristics, including security, reliability, performance efficiency, and interaction capability, ensuring the software meets rigorous quality standards and user expectations.

UML 2.5.1: UML 2.5.1 serves as the standard visual modeling language for documenting system artifacts. The team employs UML behavioral and structural diagrams to visualize the architecture, ensuring precise communication of the system's design logic among developers and stakeholders.

KVKK (Law No. 6698) The Law on the Protection of Personal Data (KVKK No. 6698) establishes the legal framework for data privacy in Turkey. The system ensures compliance by adhering to strict obligations regarding explicit consent and data security, implementing necessary technical and administrative measures to guarantee lawful processing and the protection of personal rights.

Object-Oriented Design Principles: Object-Oriented Design (OOD) principles guide the system's architecture to ensure modularity and maintainability. By adhering to the SOLID principles and utilizing encapsulation and polymorphism, the design achieves a separation of concerns that facilitates future scalability and code reuse.

4.2 Risks and Alternatives

4.2.1 Risk of Data Breach

Main risks include data breaches and privacy violations, as the application hosts highly sensitive personal thoughts, emotional patterns, and behavioral information. A security breach, via weak practices, poor configuration, or external attacks, might lead to compromised user trust and serious consequences, as the data is of a very personal nature. In order to mitigate such risks, the system should implement robust authentication, encryption, and access-control mechanisms, adhere to KVKK/GDPR requirements, and implement strict data-handling policies. It should also regularly conduct security audits, perform penetration testing, and

practice active monitoring to become aware of any vulnerabilities early on and keep user data secure.

4.2.2 Risk of Users Experiencing Severe Emotional Distress

The significant risk here is that some users may indeed be in a state of deep emotional crisis, possibly depressed or suicidal, and may use the app when professional support is most needed. Because the system is not a clinical tool and cannot treat users therapeutically, it is possible that vulnerable users may misunderstand the purpose of such applications or believe that they receive support from such tools when that is beyond their capability. Isolation or feelings of hopelessness could follow. In order to mitigate this risk, the application should communicate clearly that it is not professional care, should include crisis-support messages if high-risk language is detected, and should route users to emergency hotlines or mental health professionals when needed to make sure the system is responding safely and in a responsible way to signals of crisis.

4.3 Project Plan

WP#	Work package title	Leader	Members involved
WP1	Project Specification Report	Deniz Yazıcı	Mehmet Eren Anbar, Furkan Mert Aksakal, Deniz Şahin, Ali Deniz Sözer, Deniz Yazıcı
WP2	Analysis and Requirements Report	Furkan Mert Aksakal	Mehmet Eren Anbar, Furkan Mert Aksakal, Deniz Şahin, Ali Deniz Sözer, Deniz Yazıcı
WP3	Detailed Design Report	Deniz Şahin	Mehmet Eren Anbar, Furkan Mert Aksakal, Deniz Şahin, Ali Deniz Sözer, Deniz Yazıcı
WP4	Final Report	Ali Deniz Sözer	Mehmet Eren Anbar, Furkan Mert Aksakal, Deniz Şahin, Ali Deniz Sözer, Deniz Yazıcı
WP5	Presentation and Prototype Demo	Mehmet Eren Anbar	Mehmet Eren Anbar, Furkan Mert Aksakal, Deniz Şahin, Ali Deniz Sözer, Deniz Yazıcı
WP6	UI Design	Ali Deniz Sözer	Ali Deniz Sözer, Deniz Yazıcı
WP7	Frontend	Deniz Şahin	Deniz Şahin, Ali Deniz Sözer, Deniz Yazıcı
WP8	Backend	Deniz Yazıcı	Mehmet Eren Anbar, Furkan Mert Aksakal, Deniz Şahin, Ali Deniz Sözer, Deniz Yazıcı
WP9	Large Language Model (Agent)	Furkan Mert Aksakal	Mehmet Eren Anbar, Furkan Mert Aksakal

WP10	Project Management	Ali Deniz Sözer	Mehmet Eren Anbar, Furkan Mert Aksakal, Deniz Şahin, Ali Deniz Sözer, Deniz Yazıcı
WP11	Presentation and Final Demo	Mehmet Eren Anbar	Mehmet Eren Anbar, Furkan Mert Aksakal, Deniz Şahin, Ali Deniz Sözer, Deniz Yazıcı
WP12	Security	Deniz Şahin	Deniz Şahin

Table 1. Work Packages

WP 1: *Project Specification Report*			
Start date: *16.11.2025* **End date:** *28.11.2025*			
Leader:	*Deniz Yazıcı*	**Members involved:**	Mehmet Eren Anbar, Furkan Mert Aksakal, Deniz Şahin, Ali Deniz Sözer, Deniz Yazıcı
Objectives: *This work package indicates the plan to prepare a specification report.*			
Tasks: ***Task 1.1 Introduction*** ***Task 1.2 Description*** ***Task 1.3 High Level System Architecture & Components of Proposed Solution*** ***Task 1.4 Constraints*** ***Task 1.5 Professional and Ethical Issues*** ***Task 1.6 Standards*** ***Task 1.7 Design Requirements*** ***Task 1.8 Functional Requirements*** ***Task 1.9 Non-Functional Requirements*** ***Task 1.10 Feasibility Discussions***			
Deliverables ***D1.1:*** Project Specification Report			

Table 2. Work Package 1

WP 2: *Analysis and Requirements Report*			
Start date: *10.12.2025* **End date:** *18.12.2025*			
Leader:	*Furkan Mert Aksakal*	**Members involved:**	Mehmet Eren Anbar, Furkan Mert Aksakal, Deniz Şahin, Ali Deniz Sözer, Deniz Yazıcı
Objectives: *This work package indicates the plan to prepare analysis and requirements report.*			
Tasks: ***Task 2.1 Use-Case Model*** ***Task 2.2 Object and Class Model*** ***Task 2.3 Dynamic Model*** ***Task 2.4 Sequence Diagrams*** ***Task 2.5 Mock-up UI Diagrams*** ***Task 2.6 Activity Diagrams*** ***Task 2.7 State Diagrams*** ***Task 2.8 Explanation of Flow*** ***Task 2.9 Constraints*** ***Task 2.10 Risks and Alternatives*** ***Task 2.11 Project Plan***			
Deliverables ***D2.1:*** *Analysis and Requirements Report*			

Table 3. Work Package 2

WP 3: *Detailed Design Report*			
Start date: *01.03.2026* **End date:** *10.03.2026*			
Leader:	*Deniz Şahin*	**Members involved:**	Mehmet Eren Anbar, Furkan Mert Aksakal, Deniz Şahin, Ali Deniz Sözer, Deniz Yazıcı
Objectives: *This work package indicates the plan to prepare a detailed design report.*			
Tasks: ***Task 3.1 Introduction*** ***Task 3.2 Current System Architecture***			

Task 3.3 Proposed Software Architecture
Task 3.4 Subsystem Services
Task 3.5 Test Cases
Task 3.6 Considerations of Various Factors in Engineering Design
Task 3.7 Teamwork Details
Task 3.8 Conclusion
Deliverables
D3.1: Detailed Design Report

Table 4. Work Package 3

WP 4: *Final Report*			
Start date: *Second Semester* **End date:** *Second Semester*			
Leader:	*Ali Deniz Sözer*	**Members involved:**	Mehmet Eren Anbar, Furkan Mert Aksakal, Deniz Şahin, Ali Deniz Sözer, Deniz Yazıcı
Objectives: *This work package indicates the plan to prepare a final report*			
Tasks: ***4.1 Introduction*** ***4.2 Requirements Details*** ***4.3 Final Architecture and Design Details*** ***4.4 Development/Implementation Details*** ***4.5 Test Cases and Results*** ***4.6 Maintenance Plan and Details*** ***4.7 Other Project Elements*** ***4.7.1 Consideration of Various Factors in Engineering Design*** ***4.7.2 Constraints*** ***4.7.3 Standards*** ***4.7.1 Ethics and Professional Responsibilities*** ***4.7.2. Teamwork Details*** ***4.8.3 Conclusion and Future Work***			
Deliverables ***D4.1:*** Final Report			

Table 5. Work Package 4

WP 5: *Presentation and Prototype Demo*			
Start date: *15.12.2025* **End date:** *22.12.2025*			
Leader:	*Mehmet Eren Anbar*	**Members involved:**	Mehmet Eren Anbar, Furkan Mert Aksakal, Deniz Şahin, Ali Deniz Sözer, Deniz Yazıcı
Objectives: *This work package indicates what to demonstrate as product functionality.*			
Tasks: ***Task 5.1 Create presentation flow, who will demonstrate which part in what order.*** ***Task 5.2 Demonstrate creating an account.*** ***Task 5.3 Demonstrate journaling, messaging with the chatbot*** ***Task 5.4 Demonstrate the mind map visual consisting of various clusters and thoughts.***			
Deliverables ***D1.1:*** Prototype Demo			

Table 6. Work Package 5

WP 6: *UI Design*			
Start date: *4.12.2025* **End date:** *17.12.2025*			
Leader:	*Ali Deniz Sözer*	**Members involved:**	Mehmet Eren Anbar, Furkan Mert Aksakal, Deniz Şahin, Ali Deniz Sözer, Deniz Yazıcı
Objectives: *The designing of the UI components of the app*			
Tasks: ***Task 6.1 Login Page*** ***Task 6.2 Register Page*** ***Task 6.3 Journal Page*** ***Task 6.4 Session Page*** ***Task 6.5 Conversation Page*** ***Task 6.6 Mind Map Page*** ***Task 6.7 Stats Page*** ***Task 6.8 Actions Page*** ***Task 6.9 Profile Page*** ***Task 6.10 Landing Page*** ***Task 6.11 Onboarding Page***			

Deliverables
D6.1: UI Mock-up Diagrams

Table 7. Work Package 6

WP 7: *Frontend*			
Start date: *03.12.2025* **End date:** *30.04.2026*			
Leader:	*Deniz Şahin*	**Members involved:**	Mehmet Eren Anbar, Furkan Mert Aksakal, Deniz Şahin, Ali Deniz Sözer, Deniz Yazıcı
Objectives: *This work package indicates the work to be done for the frontend.*			
Tasks: ***Task 7.1 Map out the needed pages*** ***Task 7.2 Create drawings of pages*** ***Task 7.3 Code rough outlines of pages*** ***Task 7.4 Iterate over the pages to improve visuals and movement*** ***Task 7.5 Test pages and resolve any bugs*** ***(Possible) Task 7.6 Improve UI based on feedback from users*** ***(Possible) Task 7.7 Improve UI based on UI expert feedback***			
Deliverables ***D7.1:*** Demo 1 (December 2025) **D7.2:** Demo 2 (May 2026)			

Table 8. Work Package 7

WP 8: *Backend*			
Start date: *03.11.2025* **End date:** *18.05.2026*			
Leader:	*Deniz Yazıcı*	**Members involved:**	Mehmet Eren Anbar, Furkan Mert Aksakal, Deniz Şahin, Ali Deniz Sözer, Deniz Yazıcı
Objectives: *This work package indicates the plan to develop the backend services for the MindJournal project.*			
Tasks: ***Task 8.1.1 Authentication & Account Management*** ***Task 8.1.2 User Profile & Settings*** ***Task 8.2.1 Database Initialization & Migrations*** ***Task 8.2.2 Core Data Models*** ***Task 8.3.1 Journaling Session API*** ***Task 8.3.2 Thought Submission API***			

Task 8.4.1 AI/NLP Integration Layer ***Task 8.4.2 Thought Classification Service*** ***Task 8.4.3 CBT Output Service*** ***Task 8.4.4 Mind-Map Generation Service*** ***Task 8.5.1 Statistics & Insights Service*** ***Task 8.6.1 Security & Compliance*** ***Task 8.7.1 Testing***
Deliverables ***D8.1:*** Mindjournal Backend

Table 9. Work Package 8

WP 9: *Large Language Model (Agent)*			
Start date: *15.10.2025* **End date:** *18.05.2026*			
Leader:	*Furkan Mert Aksakal*	**Members involved:**	Mehmet Eren Anbar, Furkan Mert Aksakal, Deniz Şahin, Ali Deniz Sözer, Deniz Yazıcı
Objectives: *This work package indicates the plan to prepare a specification report.*			
Tasks: ***Task 9.1.1 Journal Agent – Guiding Question Generation*** ***Task 9.1.2 Journal Agent – Conversation State Management*** ***Task 9.1.3 Journal Agent – Thought Extraction & Normalization*** ***Task 9.2.1 Categorizer Agent – Thought Classification Pipeline*** ***Task 9.2.2 Categorizer Agent – Fast Category Check*** ***Task 9.2.3 Categorizer Agent – Tagging & Metadata Enrichment*** ***Task 9.3.1 CBT Agent – Thought Reframing Suggestions***			
Deliverables ***D9.1: Journal Agent*** ***D9.2:* Categorizer Agent** ***D9.3:* CBT Agent**			

Table 10. Work Package 9

WP 10: *Project Management*			
Start date: *September 2025* **End date:** *May 2026*			
Leader:	*Ali Deniz Sözer*	**Members involved:**	Mehmet Eren Anbar, Furkan Mert Aksakal, Deniz Şahin, Ali Deniz Sözer, Deniz Yazıcı
Objectives: *This work package indicates the project management phase*			
Tasks: No specific predefined tasks for this work plan			

Deliverables: No specific deliverables for this work plan

Table 11. Work Package 10

WP 11: *Present and Final Demo*			
Start date: *01.05.2026* **End date:** *18.05.2026*			
Leader:	Mehmet Eren Anbar	**Members involved:**	Mehmet Eren Anbar, Furkan Mert Aksakal, Deniz Şahin, Ali Deniz Sözer, Deniz Yazıcı
Objectives: *This work package indicates how the final presentation and demo will be made.*			
Tasks: ***Task 11.1 Create presentation flow, who will demonstrate which part in what order.*** ***Task 11.2 Demonstrate creating an account.*** ***Task 11.3 Demonstrate journaling, messaging with the chatbot*** ***Task 11.4 Demonstrate the mind map visual consisting of various clusters and thoughts.***			
Deliverables ***D11.1:*** Demonstration of the working product			

Table 12. Work Package 11

WP 12: *Security*			
Start date: *01.2026* **End date:** *05.2026*			
Leader:	Deniz Şahin	**Members involved:**	Deniz Şahin
Objectives: *This work package indicates the plan to prepare the security of the system.*			
Tasks: ***Task 12.1 Decide on location of different forms of data storage*** ***Task 12.2 Decide on encryption methods of different data stores*** ***Task 12.3 Create basic authentication system*** ***Task 12.4 Create authorization for remote servers*** ***Task 12.5 Implement data encryption*** ***Task 12.6 Ensure AI model data safety*** ***Task 12.7 Implement method for selective on device access safety*** ***Task 12.8 Pentest***			

Deliverables
D12.1: Demo 2 (May)

Table 13. Work Package 12

4.4 Ensuring Proper Teamwork

To guarantee seamless collaboration and efficient teamwork throughout the development of MindJournal, we have implemented a systematic approach that utilizes industry-standard tools and transparent communication channels. Our strategy focuses on clear role definition, consistent progress tracking, and iterative feedback loops to keep the project aligned with its goals.

We rely on GitHub Issues as our primary project management tool. By utilizing a Kanban board structure, we categorize tasks into "To Do," "In Progress," "Under Review," and "Completed," providing a real-time visual representation of the project's status. Specific tasks are assigned to individual team members via GitHub to ensure accountability.

For day-to-day communication, we use WhatsApp to handle rapid updates, query resolution, and deadline reminders. This acts as our immediate channel for addressing blockers and ensuring the team remains connected and responsive. Furthermore, after receiving feedback from supervisors and jury members during formal progress meetings, we schedule dedicated follow-up sessions online to reflect on the meetings and our progress.

4.5 Ethics and Professional Responsibilities

- The application will be working with sensitive data (people's thoughts, experiences and names of people around them etc.) the application will have to be developed with extra caution of data privacy in mind. Especially since the data people enter cannot be monitored in terms of its identifiability and therefore will have to be treated as identifiable data making it especially challenging to abide by KVKK regulations.
- Since this application will be interacting with potentially psychologically vulnerable people the messages sent by the application and the potential advice given will have to be accepted by psychologists working in the field. In case someone is advised in a poor manner the burden is on the application developers.
- The data retention aspect of the application needs to be taken into account. Users will probably end up using this application over an extended period of time in order to

understand their own thought processes. The data collected over long periods of time may (hopefully) end up being useful to someone in understanding their thoughts and ideas. For this reason extra effort should be put into not losing said data over time.

4.6 Planning for New Knowledge and Learning Strategies

For this project most of our work division is done based on our personal interests and therefore mostly coincides with our previous knowledge. Furthermore, thanks to the wide diversity of interest fields of our group ranging from artificial intelligence to computer graphics to cybersecurity, most tasks can be assigned to someone already interested in the topic. This allows for us to use a vast amount of previously acquired knowledge from prior research and elective courses we have taken. On top of this most of us have their own way of acquiring new knowledge based on the field we are working on and our personal preferences. Some of us read long documentations, some chose to try out short projects on the topic and some use AI for fast research into the depths of the topic. Since we are using our old knowledge combined with already established learning strategies, we are able to progress in the project without long delays for learning and therefore do not have specific learning strategies for this project.

5 Glossary

Affective Computing: A branch of computer science and engineering that involves the study and development of systems that can recognize, interpret, process, and simulate human affects (emotions).

Cognitive Behavioral Therapy (CBT): A psycho-social intervention that aims to improve mental health by identifying and challenging cognitive distortions (negative thought patterns) and employing behavioral strategies to regulate emotions.

Cognitive Distortion: Irrational or exaggerated thought patterns (e.g., catastrophizing, black-and-white thinking) that are believed to perpetuate the effects of psychopathological states like depression and anxiety.

Cognitive Reframing: A psychological technique used in CBT to identify negative thoughts and replace them with more balanced, rational, and positive alternatives.

DRY (Don't Repeat Yourself): A software development principle aimed at reducing repetition of software patterns, replacing it with abstractions or using data normalization to avoid redundancy.

Embodied Conversational Agent (ECA): An intelligent agent that uses a visual representation (avatar) to interact with users, utilizing both verbal (speech) and non-verbal (facial expressions, gestures) communication.

KVKK (Kişisel Verilerin Korunması Kanunu): The Turkish Law on the Protection of Personal Data (Law No. 6698), which regulates the processing of personal data and ensures the privacy rights of individuals.

Large Language Model (LLM): A type of artificial intelligence algorithm that uses deep learning techniques and massively large data sets to understand, summarize, generate, and predict new content.

Mind Map (in MindJournal context): A visual diagram used to organize information where the central concepts are user-defined categories, and branches represent specific thoughts, tailored to reveal the structure of the user's psyche.

Natural Language Processing (NLP): A subfield of artificial intelligence capable of reading, understanding, and deriving meaning from human languages.

SOLID: An acronym for five design principles (Single Responsibility, Open-Closed, Liskov Substitution, Interface Segregation, Dependency Inversion) intended to make software designs more understandable, flexible, and maintainable.

SRP (Single Responsibility Principle): A computer programming principle that states that every module, class, or function should have responsibility over a single part of the functionality provided by the software.

Thought Chain: A specific visualization feature in MindJournal that connects thoughts via directed edges to illustrate the causal or triggering relationship between them.

6 References

[1] Wysa, "Everyday Mental Health," *Wysa*. [Online]. Available: https://www.wysa.com. [Accessed: Dec. 17, 2025].

[2] Mindsera, "AI Journal for Mental Wellbeing," *Mindsera*. [Online]. Available: https://www.mindsera.com. [Accessed: Dec. 17, 2025].

[3] Reflectly, "A Journal for Happiness," *Reflectly*. [Online]. Available: https://reflectlyapp.com. [Accessed: Dec. 17, 2025].

[4] Clearful, "The Best Journal and Diary App for Personal Growth," *Clearful*. [Online]. Available: https://clearful.com. [Accessed: Dec. 17, 2025].

[5] Stoic, "Beat Stress in 120 Seconds," *Stoic*. [Online]. Available: https://www.getstoic.com. [Accessed: Dec. 17, 2025].

[6] MindDoc, "Your Mental Health Companion," *MindDoc*. [Online]. Available: https://www.minddoc.com. [Accessed: Dec. 17, 2025].

[7] Bloom, "Self-Therapy App," *Bloom*. [Online]. Available: https://www.enjoybloom.com. [Accessed: Dec. 17, 2025].

[8] Calm, "The #1 App for Meditation and Sleep," *Calm*. [Online]. Available: https://www.calm.com. [Accessed: Dec. 17, 2025].

[9] Headspace, "Mental Health App for Meditation & Sleep," *Headspace*. [Online]. Available: https://www.headspace.com. [Accessed: Dec. 17, 2025].