



International Journal of Advanced Research in Education and Technology (IJARETY)

Volume 13, Issue 3, May - June 2026

Impact Factor: 9.971



TherapEase: An AI-Powered Web-Based Mental Health Support System

Gagan M, Gnaneswari Gnanaguru

Department of Master of Computer Applications (MCA), CMR Institute of Technology, Bengaluru, Karnataka, India

Associate Professor, Department of Master of Computer Applications, CMR Institute of Technology, Bengaluru,

Karnataka, India

ABSTRACT: Access to mental health care remains a significant challenge in modern society due to geographic barriers, high service costs, social stigma, and limited availability of timely professional support. This paper presents TherapEase, an artificial intelligence-powered, web-based mental health support platform designed to provide accessible, conversational emotional assistance to individuals in need. The system leverages modern web technologies including Next.js, React, TypeScript, and Tailwind CSS on the frontend, alongside Node.js and the Groq API for AI-driven natural language processing on the backend. TherapEase implements a modular architecture that integrates secure user authentication via Clerk, configurable AI model settings, and a responsive user interface supporting multiple features including AI chat, mood tracking, relaxation exercises, appointment booking, and medication reminders. The system prioritizes user privacy, data security, and ethical AI usage, explicitly positioning itself as an early-intervention supplement to—not a replacement for—professional therapy. Comprehensive testing across functional, non-functional, boundary, security, and performance categories demonstrates the system’s reliability and readiness for deployment.

KEYWORDS: Mental Health, Artificial Intelligence, Chatbot, Web Application, Natural Language Processing, Next.js, Groq API, Clerk Authentication, Emotional Support System

I. INTRODUCTION

Mental health has emerged as a critical component of overall human well-being in the twenty-first century. Rising stress levels, social isolation, and fast-paced modern lifestyles have intensified the demand for accessible emotional support services. Despite growing awareness, a significant treatment gap persists—driven by prohibitive costs, geographical constraints, long waiting lists, and pervasive social stigma that discourages many individuals from seeking help [1].

Digital technologies present a promising avenue to bridge this gap. Smartphone penetration, ubiquitous internet access, and advances in artificial intelligence have collectively enabled the development of scalable, low-cost mental health tools that can reach users regardless of location or time of day. AI-driven chatbots and conversational agents, in particular, offer an empathetic, always-available interaction model well-suited to early emotional support [4].

TherapEase is an AI-powered web application developed to address these challenges. Built on a modular, component-based architecture, the platform delivers real-time conversational support, mood monitoring, relaxation guidance, and health management features through an intuitive browser-based interface. The system is engineered with security and ethical responsibility at its core, ensuring that user data remains protected and that the platform’s role is appropriately scoped to supportive guidance rather than clinical diagnosis.

This paper is organized as follows: Section II reviews related literature. Section III details the system architecture and technology stack. Section IV describes software requirements. Section V presents system design. Section VI covers implementation. Section VII reports testing results, followed by conclusions and future work.

II. LITERATURE REVIEW

The evolution of digital mental health support can be traced from static, informational web pages to sophisticated AI-driven conversational agents. Early platforms provided psychoeducational content in a one-way format, offering limited personalization and no real-time interaction. While such resources improved mental health literacy, they failed to address users' individual circumstances or provide responsive emotional engagement [2].

The introduction of rule-based chatbots marked a significant step forward, enabling scripted dialogue with users seeking emotional support. However, these systems were constrained by rigid decision trees that could not accommodate natural language variation or nuanced emotional expression. Responses that failed to match pre-defined patterns often resulted in generic, unhelpful outputs that eroded user trust.

Recent research has explored the integration of machine learning and large language models (LLMs) into mental health applications [4], [5]. Studies demonstrate that LLM-based conversational agents can produce contextually aware, emotionally resonant responses and maintain coherent multi-turn dialogues. Nevertheless, challenges persist around data privacy, algorithmic bias, and the ethical boundaries of AI-delivered care.

From a technical standpoint, modern web frameworks have enabled the development of scalable, device-agnostic platforms with robust security features. React-based single-page applications, server-side rendering via Next.js, and API-driven backend architectures have become standard for high-performance web platforms [9]. Identity management solutions such as Clerk provide enterprise-grade authentication without requiring custom security engineering [7].

A critical gap identified across existing literature is the absence of a unified platform that seamlessly combines adaptive AI conversation, strong security infrastructure, intuitive user experience design, and modular extensibility. TherapEase addresses this gap by integrating these dimensions within a single, cohesive system.

A. Comparison: Existing vs. Proposed System

Table 1 contrasts limitations of traditional digital mental health systems with the capabilities offered by TherapEase.

Table 1: Comparison Between Existing and Proposed Systems

Feature	Existing Systems	TherapEase
Accessibility	Limited by location & availability	24/7 access via any browser
Interaction	Static or rule-based chatbots	AI-driven, context-aware
Personalization	Minimal or none	Sentiment-aware responses
Security	Weak or inconsistent	Clerk auth, env-based config
User Experience	Complex or outdated	Responsive & intuitive
Scalability	Tightly coupled	Modular, easy upgrades

III. SYSTEM ARCHITECTURE AND TECHNOLOGY STACK

TherapEase follows a client-server architecture in which the frontend handles user interaction and presentation while the backend manages business logic, AI processing, and security. The system is entirely browser-based, requiring no local installation and supporting access across desktop, tablet, and mobile devices.

A. Frontend Technologies

Next.js serves as the primary application framework, providing server-side rendering, optimized page routing, and a component-based structure. React powers the dynamic user interface, enabling efficient DOM updates without full page reloads. TypeScript enforces strict type checking throughout the codebase, reducing runtime errors and improving maintainability. Tailwind CSS provides a utility-first styling approach that ensures visual consistency and responsive design across all screen sizes.

B. Backend and AI Technologies

Node.js provides the server runtime environment, supporting asynchronous request handling and efficient multi-user operation. The Groq API [6] is the core AI integration point, enabling real-time natural language understanding and response generation through high-performance large language models. API keys and environment-specific settings are managed through .env configuration files, preventing credential exposure in source code.

C. Authentication and Security

Clerk Authentication [7] manages all user identity operations including registration, login, session management, and logout. This integration ensures that only verified users can access the platform, protecting sensitive interactions from unauthorized access.

D. Technology Summary

Table 2 summarizes the complete technology stack.

Table 2: Technology Stack Summary

Layer	Technology	Purpose
Frontend	Next.js + React	Rendering, routing, components
Type System	TypeScript	Type safety, error detection
Styling	Tailwind CSS	Responsive UI design
Backend Runtime	Node.js	Server-side logic & APIs
AI Engine	Groq API	NLP and response generation
Authentication	Clerk	Identity & session management
Config	.env files	Secure credential storage
Code Quality	ESLint	Static analysis

IV. SOFTWARE REQUIREMENTS SPECIFICATION**A. Functional Requirements**

The TherapEase system must fulfill the following core requirements:

- **User Authentication:** Secure registration, login, session management, and logout for all users.
- **Conversational AI Interface:** Users submit text input and receive real-time AI-generated responses.
- **Mood Tracking:** Users log mood entries with scores, labels, and notes; the system visualizes trends over time.
- **Relaxation Exercises:** Guided breathing exercises, meditation sessions, and calming audio content.
- **Appointment Booking:** Users schedule sessions with qualified mental health professionals.
- **Medication Reminders:** Users register medications with dosage schedules and receive tracking support.
- **AI Model Configuration:** Administrators update AI model settings without modifying core application logic.
- **Responsive Interface:** Consistent functionality across desktop, tablet, and mobile devices.

B. Non-Functional Requirements

- **Performance:** Chatbot responses delivered within acceptable latency under normal and peak loads.
- **Scalability:** Modular architecture supports incremental feature addition and increased user volume.
- **Security:** All user data and interactions protected through authentication controls and secure configuration.
- **Reliability:** High availability with graceful error recovery.
- **Usability:** Interface navigable by users with minimal technical expertise.
- **Maintainability:** Independent module updates without system-wide disruption.

C. Hardware Requirements

Table 3 lists the minimum and recommended hardware specifications.

Table 3: Hardware Requirements

Component	Minimum	Recommended
CPU	Intel Core i3, 2.0 GHz	Core i5/i7, 3.0 GHz+
RAM	4 GB	8 GB or higher
Storage	10 GB free	20 GB SSD
Network	5 Mbps stable	20 Mbps+

V. SYSTEM DESIGN**A. System Perspective**

TherapEase operates as a standalone web application following a client-server model. The frontend, served through a web browser, handles all user interaction and visual presentation. Core application logic—including authentication, AI processing, and data management—resides on the server side, ensuring centralized control and improved security. The system interacts with external services (Clerk for authentication, Groq for AI) through secure API integrations managed via environment configuration files.

The modular design cleanly separates the presentation layer, business logic layer, and configuration layer. This separation improves reliability and enables independent modification of each component.

B. System Actors

The system serves three primary actors:

1. **Users** — individuals seeking mental health support who interact with the AI chatbot, mood tracker, and wellness features.
2. **Therapists/Counselors** — professionals who access a separate dashboard to review interactions and provide guidance-based messages.
3. **Administrators** — responsible for platform configuration, user account management, and system monitoring.

C. Data Model

The entity-relationship model defines the following core entities:

- **User** (user_id, name, email, clerk_user_id): Central entity linked to all user-specific data.
- **MoodEntry** (id, user_id, mood_label, mood_score, emotion_confidence, notes, created_at): Periodic mood logs.
- **ChatSession** (id, user_id, started_at): Groups messages into discrete sessions.
- **ChatMessage** (id, session_id, sender, message, created_at): Individual messages within a session.
- **Medication** (id, user_id, medicine_name, dosage, frequency, start_date, end_date): Medication records.
- **Appointment** (id, user_id, doctor_name, specialization, date, time, mode, status): Scheduled sessions.

Relationships follow a one-to-many pattern: one User has many MoodEntries, ChatSessions, Medications, and Appointments; one ChatSession has many ChatMessages.

VI. IMPLEMENTATION**A. Authentication Module**

User registration and sign-in are implemented using Clerk's SignUp and SignIn React components. Clerk handles all underlying security operations including password hashing, session token management, and multi-device session tracking. Protected routes redirect unauthenticated users to the sign-in page, preventing unauthorized access to any dashboard features.

B. AI Response Generation

The message generation module is implemented as a Next.js API route that accepts user messages and optional chat history context. The module performs two parallel operations: (1) generating a conversational response via the Groq API, and (2) conducting sentiment analysis on the user's input. Based on the detected emotional state, responses are dynamically enhanced with empathetic preambles. For example, sadness is acknowledged with a validating statement, anxiety triggers grounding cues, and positive emotions receive reinforcing affirmations. Approximately 30% of responses are appended with reflective prompts to encourage continued engagement. The API route accepts the following JSON payload:

```
{ "message": "<user text>", "chatHistory": [<last 5 messages>] }
```

And returns:

```
{
  "response": "<enhanced AI reply>",
  "sentiment": {
    "sentiment": "positive|negative|neutral",
    "confidence": 0.85,
    "emotion": "happy"
  },
  "timestamp": "<ISO 8601>"
}
```

C. Chat Interface

The ChatInterface component maintains full conversation state using React's useState hook and persists chat history to localStorage for continuity across sessions. Messages are rendered in a scrollable container with timestamps and sentiment labels. Bot messages include action controls for liking, disliking, and copying responses—enabling implicit feedback collection.

D. Dashboard Navigation

The main page renders a persistent sidebar listing five feature modules: AI Chat, Mood Tracker, Relaxation Exercises, Appointments, and Medication Reminders. Active tab state controls which module is rendered in the main content area. The layout adapts gracefully to different viewport sizes, ensuring accessibility on mobile devices.

E. Security Implementation

Security is enforced at multiple layers:

- Authentication gates restrict dashboard access to verified users.
- API routes validate input types and formats before processing.
- Environment variables store all sensitive credentials and are never exposed in client-side code.
- Modular backend structure limits the blast radius of any single component failure.

VII. TESTING

A. Test Strategy

Testing was conducted across five categories: functional, non-functional, boundary case, security, and performance. The goal was to verify correctness, robustness, and safety across all use cases and edge conditions.

B. Functional Test Results

Table 4 presents the functional test case results.

Table 4: Functional Test Cases

ID	Description	Expected Output	Result
TC01	Login – valid credentials	Dashboard displayed	Pass
TC02	Login – invalid credentials	Error message shown	Pass
TC03	Access chatbot dashboard	Chat interface opens	Pass
TC04	Send message to chatbot	Message delivered	Pass
TC05	Chatbot response generation	AI response received	Pass
TC06	Display chatbot response	Response shown	Pass
TC07	Message storage	Stored in localStorage	Pass
TC08	Logout functionality	Session terminated	Pass

C. Boundary Testing

Empty messages were correctly blocked from submission. Messages exceeding 500 characters were processed without errors. Rapidly submitted sequential messages were handled with minimal delay. Multiple concurrent user sessions operated independently without data cross-contamination.

D. Security Testing

Unauthorized URL access was successfully blocked by authentication middleware. SQL injection attempts were neutralized by parameterized query handling. Session manipulation attempts via cache clearing were detected and access was denied. Message tampering during transmission was prevented by secure API communication channels.

E. Performance Testing

Chatbot responses were delivered within acceptable time limits under normal load. Multi-user concurrent access produced no measurable performance degradation. Database read/write operations completed without noticeable latency. System stability was maintained throughout extended usage periods.

VIII. CONCLUSION

This paper presented TherapEase, an AI-powered web-based mental health support system designed to make emotional assistance more accessible, responsive, and trustworthy. By combining a modular Next.js architecture with Groq API-driven natural language processing, Clerk-based authentication, and a thoughtfully designed multi-feature dashboard, TherapEase demonstrates how modern web technologies can meaningfully extend mental health support to underserved populations.

The system successfully achieves its design objectives: providing real-time conversational support, ensuring user privacy through robust security practices, maintaining high usability across device types, and enabling future extensibility through its modular architecture. All eight functional test cases passed, and boundary, security, and performance evaluations confirmed the system's reliability and readiness for production deployment.

Importantly, TherapEase is designed with ethical clarity—it supports and guides users toward professional care rather than attempting to replace licensed therapists. This positioning ensures that the platform operates responsibly within the emerging landscape of AI-assisted mental healthcare.

IX. FUTURE ENHANCEMENTS

- **Personalized AI Memory:** Incorporating long-term conversation history so the AI can tailor responses based on a user's established patterns and progress.
- **Voice Interaction:** Adding speech-to-text and text-to-speech capabilities to support voice-based interaction.
- **Live Therapist Integration:** A real-time video or messaging channel connecting users directly with licensed therapists.
- **Mobile Application:** Native iOS and Android applications to extend accessibility beyond web browsers.

- **Multi-Factor Authentication:** Additional security layer for sensitive data protection.
- **Multilingual Support:** Expanding language options to serve diverse user populations.
- **Advanced Analytics:** Aggregate mood trend analysis and wellness insights for users and (with consent) therapists.

REFERENCES

- [1]R. S. Pressman, Software Engineering: A Practitioner's Approach, 8th ed. New York, NY, USA: McGraw-Hill Education, 2014.
- [2]I. Sommerville, Software Engineering, 10th ed. Hoboken, NJ, USA: Pearson Education, 2015.
- [3]A. Dennis, B. H. Wixom, and R. M. Roth, Systems Analysis and Design, 6th ed. Hoboken, NJ, USA: Wiley, 2018.
- [4]A. N. Vaidyam, H. Wisniewski, J. D. Halamka, M. S. Kashavan, and J. B. Torous, "Chatbots and conversational agents in mental health: A review of the psychiatric landscape," Canadian Journal of Psychiatry, vol. 64, no. 7, pp. 456–464, 2019.
- [5]Woebot Health, "The role of AI in mental health: Accessible support through conversational agents," Woebot Health Research Report, 2021.
- [6]Groq Inc., "Groq API Reference Documentation," 2024. [Online]. Available: <https://console.groq.com/docs/api-reference>
- [7]Clerk Inc., "Clerk Authentication Documentation," 2024. [Online]. Available: <https://clerk.com/docs>
- [8]Mozilla Developer Network, "Web Development Documentation," 2024. [Online]. Available: <https://developer.mozilla.org>
- [9]Vercel Inc., "Next.js Documentation," 2024. [Online]. Available: <https://nextjs.org/docs>
- [10]Object Management Group, "Unified Modeling Language Specification," v2.5.1, OMG Document formal/2017-12-05, 2017.



International Journal of Advanced Research in Education and Technology

ISSN: 2394-2975

Impact Factor: 9.971