iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** XI **Month of publication:** November 2025

DOI: https://doi.org/10.22214/ijraset.2025.75342

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XI Nov 2025- Available at www.ijraset.com

Mindvault AI Journaling App

Diviyesh P[1], Dravid P[2], Hemachandran D[3], Mrs. K. Tamilselvi[4]
[1, 2, 3]*Bachelor of Engineering In Computer Science And Engineering, Adhiyamaan College Of Engineering (An Autonomous Institution), ANNA University, Chennai*
[4]*Assistant Professor, Department of CSE, Adhiyamaan College of Engineering (An Autonomous Institution), Anna University, Chennai*

Abstract: ***The MindVault – AI Journaling App is an innovative digital platform that integrates artificial intelligence and natural language processing to enhance the traditional journaling experience. It is designed to help individuals express their emotions, thoughts, and daily reflections while receiving meaningful insights into their mental well-being. The app allows users to log in, write journal entries, and instantly get emotional feedback generated through AI-based sentiment and mood analysis. By examining text patterns and tone, the system identifies emotions such as happiness, sadness, anxiety, or calmness and presents them visually using mood graphs and emotion summaries. This helps users track their emotional changes over time and better understand their psychological patterns. MindVault prioritizes user privacy by implementing secure authentication and encrypted data storage, ensuring a safe and confidential journaling environment. The system is developed using technologies such as Python, Flask, React.js, TensorFlow, and MySQL, making it robust, scalable, and efficient. It bridges the gap between emotional awareness and technology, providing users with a modern and interactive way to reflect on their feelings. The app not only supports mental wellness but also encourages users to cultivate mindfulness and self-awareness in their everyday lives. Future enhancements could include voice journaling, deeper emotional analysis, and AI-generated wellness recommendations. Overall, MindVault – AI Journaling App transforms journaling into an intelligent self-reflective process, empowering users to understand themselves better and maintain a healthy emotional balance in their digital journey.***

I. INTRODUCTION

A. Overview

The MindVault – AI Journaling App is an innovative digital journaling platform that leverages Artificial Intelligence (AI) to transform the traditional process of maintaining a personal diary into an intelligent, interactive, and insightful experience. In today's fast- paced world, individuals often struggle to express their emotions, organize thoughts, or maintain consistency in journaling. MindVault addresses these challenges by offering a user-friendly, cloud-based environment where users can record, analyze, and reflect on their daily experiences efficiently. The app is developed using React and TypeScript for the frontend, ensuring a smooth and dynamic user interface, while Supabase provides a secure backend for authentication and data storage. MindVault allows users to create personal accounts, write journal entries, and access them anytime through a visually appealing dashboard. Integrated AI features enable the system to summarize journal entries, detect emotional tone, and generate personalized insights that promote mindfulness, emotional well-being, and self-awareness.

Unlike conventional note-taking or diary applications, MindVault focuses on the emotional and reflective aspects of journaling. The platform provides a distraction-free interface enhanced by Tailwind CSS for aesthetic design and responsive layouts. Security is reinforced through Supabase authentication, ensuring that all personal entries remain private and protected.

Overall, the MindVault – AI Journaling App bridges the gap between technology and emotional health by combining modern web development tools with AI-driven analytics. It serves as a valuable tool for students, professionals, and anyone seeking self-improvement and emotional balance through intelligent journaling.

B. Objective

The primary objective of the MindVault – AI Journaling App is to design and develop an intelligent digital platform that allows users to document, reflect upon, and analyze their personal thoughts and daily activities in a secure and meaningful way. MindVault aims to solve this by integrating Artificial Intelligence to assist users in writing and understanding their emotions more effectively.

One of the core objectives of this project is to build a responsive and user-friendly interface using React and TypeScript, ensuring smooth navigation and an appealing visual experience. The inclusion of Tailwind CSS further enhances the application's visual aesthetics, providing customizable themes that allow users to switch between light and dark modes according to their preferences.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XI Nov 2025- Available at www.ijraset.com

Another important goal is to ensure data security and privacy through the use of Supabase for authentication and database management. As journaling involves personal and emotional information, it is essential that all user data is stored securely in the cloud with proper access controls. MindVault provides each user with a protected account where their entries are saved privately, accessible only to them, thus creating a trustworthy environment for open self-expression.

Furthermore, MindVault seeks to demonstrate the effective use of modern web technologies to build scalable, cloud-based applications that serve both functional and emotional needs. The long-term vision of the project is to evolve into a personal digital companion that can analyze growth patterns, suggest positive affirmations, and encourage users to build a consistent habit of reflection. Through these objectives, the MindVault – AI Journaling App stands as a step forward in merging technology, psychology, and creativity to foster better mental health and personal development in the digital era.

II. LITERATURE SURVEY

A. Background and relevance

Journaling as a practice has been documented across psychology and wellness literature for its benefits in emotional regulation, memory consolidation, and personal insight. In the digital age, journaling applications have shifted this activity from paper to screens, adding convenience and new possibilities such as searchability, multimedia entries, and cross- device sync. Recent research and product work emphasize not only storing entries but also extracting value from them — for example, through automated summarization, trend detection, and mood tracking. This body of work sets the foundation for applications that combine therapeutic goals with computational assistance.

B. Existing digital journaling tools

A number of commercial and open-source journaling platforms have explored features like tagging, reminders, multimedia support, and calendar views to encourage regular writing. These products demonstrate common UI/UX patterns such as distraction-free editors, chronological organization, and privacy-first design. However, most mainstream apps stop short of deeply analyzing user content — offering at best simple statistics (word counts, streaks) or manual tagging — leaving room for richer intelligence-driven features that can provide actionable insight without compromising usability.

C. AI-driven text analysis techniques

Natural language processing (NLP) methods such as extractive and abstractive summarization, sentiment analysis, topic modeling, and named-entity recognition have matured rapidly. Summarization algorithms help condense long entries into concise takeaways; sentiment and emotion detection models estimate affective tone over single entries or across time; topic models and clustering can surface recurring themes (e.g., work, relationships, health). Recent advances in transformer-based language models enable more context-aware outputs, improving relevance and readability of generated summaries and insights.

D. Applications of AI in mental health and reflective tools

Research integrating AI with mental health tools highlights both promise and caution: automated feedback can help users recognize patterns and prompt reflection, yet overly prescriptive or inaccurate suggestions may harm engagement or trust. Successful approaches combine lightweight, explainable outputs with clear user control — for instance, offering suggested summaries and tags that users can edit, rather than fully automated changes. Ethical studies stress transparency, opt-in analytics, and safeguards against misinterpretation of sensitive content.

E. Privacy, security, and ethical considerations

Because journal entries often contain intimate personal data, the literature strongly recommends robust privacy-by-design principles: end-to-end encryption or secure cloud storage, strict access controls, minimal data retention policies, and clear, user-friendly consent flows. Works in human–computer interaction argue for giving users clear visibility into what AI sees and does (e.g., showing highlight snippets used to generate insights) to build trust and autonomy.

F. UX research findings

Usability studies of journaling apps show that people value simplicity, low friction, and gentle reminders. Features that support reflection — such as prompts, visualizations of mood over time, and the ability to revisit past entries easily — increase long-term adherence.

Importantly, users prefer personalization and control: customizable prompts, theme settings, and an ability to opt-out of analytics increase acceptance.

G. *Technical gaps and motivation for MindVault*

Despite advances, gaps remain: many apps lack deep yet user-controllable AI insights; few combine strong privacy guarantees with on-device or minimally invasive cloud AI; and integration of mood analytics, summarization, and long-term trend visualization in a single, elegant interface is uncommon. These gaps motivate MindVault's focus on integrating summarization, emotion detection, and trend visualization while emphasizing security and user agency.

H. *Summary and direction*

The literature supports a design that balances automated intelligence with transparency, prioritizes privacy, and focuses on low-friction, reflective UX. MindVault builds on proven NLP techniques and best-practice UX and privacy principles to offer an AI-enhanced journaling experience aimed at improving self-awareness and habit formation without sacrificing user trust.

III. SYSTEM ANALYSIS

A. *Existing System*

In the existing system, most journaling applications act as basic digital diaries that allow users to write, save, and access their daily thoughts. Apps like Google Keep, Evernote, and Day One Journal make note-taking convenient, but their role ends at storage and organization. These tools lack intelligence or emotional awareness, turning journaling into a simple documentation process rather than a reflective or interactive experience.

Users can record their feelings or daily events, but the system provides no feedback or insight. There is no emotional analysis, mood tracking, or writing suggestion, which reduces engagement and long-term interest. Moreover, the absence of AI and Natural Language Processing (NLP) means these platforms cannot analyze emotions, summarize entries, or generate personalized insights to help users understand their mental or emotional state.

Data privacy is another issue in the existing system. Many apps store information on third- party servers without clear encryption or privacy control, which can lead to concerns about the safety of personal journal data.

Limitations of Existing System

1) No AI Integration: Cannot analyze or summarize journal entries.
2) Low User Engagement: Journaling remains passive and uninteractive.
3) Limited Personalization: Lacks mood tracking or adaptive suggestions.
4) Privacy Issues: Weak protection for personal and emotional data.
5) No Emotional Support: Fails to promote mindfulness or self-awareness.

B. *Proposed System*

The proposed system, MindVault – AI Journaling App, introduces an intelligent and emotionally aware journaling platform that enhances self-reflection and mental well- being. Unlike traditional journaling tools that only store entries, MindVault uses Artificial Intelligence (AI) and Natural Language Processing (NLP) to analyze user writings, identify emotional tones, and provide meaningful insights.

MindVault allows users to write freely about their daily thoughts and emotions while the AI automatically detects mood patterns, summarizes content, and offers reflective prompts or motivational feedback. It also provides personalized suggestions to encourage positive thinking and consistent journaling habits.

To ensure security, the system incorporates data encryption and user-controlled privacy settings. The design emphasizes a user-friendly interface with customizable themes and writing modes to create a calm and inspiring journaling experience.

Advantages of Proposed System

1) AI and NLP Integration: Analyzes text to detect emotions, summarize entries, and give personalized insights.
2) Enhanced User Engagement: Interactive feedback encourages regular journaling and reflection.
3) Strong Data Privacy: End-to-end encryption ensures the safety of personal information.
4) Personalized Experience**:** Customizable interface, themes, and writing prompts tailored to user preferences.

In summary, MindVault transforms journaling from a simple writing task into a smart, secure experience that promotes mindfulness and self-growth.

C. Proposed Solution

The proposed solution for MindVault – AI Journaling App focuses on transforming the traditional journaling process into a more intelligent, interactive, and emotionally supportive experience. Unlike existing systems that merely allow users to write and store entries, this solution uses Artificial Intelligence (AI) and Natural Language Processing (NLP) to understand the user's emotions and provide meaningful feedback. The system analyzes the text entered by the user, detects mood and sentiment, and offers reflective insights or motivational prompts that encourage emotional awareness and personal growth.

The app aims to create a personalized journaling experience by adapting to each user's writing style, emotional tone, and preferences. Through AI-generated mood tracking and visual reports, users can view their emotional patterns over time, helping them understand their thoughts and behaviors better. This feature makes journaling not only an act of writing but also a journey of self-discovery and mindfulness. To address data security concerns, the proposed solution incorporates end-to-end encryption and secure cloud storage, ensuring that journal entries remain private and protected. Users have full control over their data, including the ability to manage access and backups. The interface is simple, elegant, and customizable, with options for voice journaling, writing reminders, and theme selection to enhance engagement.

Overall, the proposed solution transforms MindVault into a secure, intelligent, and emotionally aware journaling platform that blends technology with mental wellness, helping users reflect, grow, and maintain a positive mindset.

D. Ideation & Brainstorming

The ideation and brainstorming phase of the MindVault – AI Journaling App was a crucial part of the system design process. The goal was to move beyond traditional note- taking by creating an intelligent and emotionally aware journaling platform that promotes mindfulness and self-reflection. During the initial brainstorming sessions, the development team analyzed the shortcomings of existing journaling apps and identified opportunities for improvement. The discussions centered on how Artificial Intelligence (AI) and Natural Language Processing (NLP) could applied to provide emotional analysis, smart recommendations, and user engagement features. Several ideas were proposed, evaluated, and refined foundation of MindVault's core functionalities.

Key Ideas Generated During Brainstorming

1) AI-Powered Emotion Analysis: To detect the mood and sentiment behind user entries.
2) Personalized Suggestions: Provide reflective prompts and motivational feedback based on writing tone.
3) Mood Tracking and Visualization: Display emotional trends through charts and reports for self-awareness.
4) Secure Data Management: Implement end-to-end encryption and user-controlled data access.
5) Voice Journaling and Reminders: Introduce voice input and daily notifications to improve consistency.
6) Cloud Synchronization: Ensure access across multiple devices with secure storage.

E. Problem-Solution Fit

1) Problem

In today's fast-paced digital life, people struggle to manage their emotions, reflect on their thoughts, and maintain mental well-being. Traditional journaling methods, whether on paper or simple note-taking apps, only allow users to record their feelings but provide no emotional insight, motivation, or guidance. Most existing journaling platforms lack intelligence, personalization, and emotional understanding

2) Solution

The MindVault – AI Journaling App solves this problem by introducing an intelligent, interactive, and secure journaling platform powered by Artificial Intelligence (AI) and Natural Language Processing (NLP). It visualizes mood trends, offers motivational prompts, and personalizes the journaling experience based on user behavior. In addition, it ensures end-to-end data encryption and user-controlled privacy, allowing users to safely store and revisit their personal reflections anytime.

This system thus perfectly fits the problem by offering:

1) Automation: AI and NLP analyze entries and provide emotional insights automatically.
2) Accessibility: Available as a user-friendly app for journaling anytime, anywhere.
3) Insight: Offers personalized feedback, summaries, and mood analysis for self- awareness.
4) Security: Uses encryption to protect personal and emotional data.
5) Engagement: Encourages consistent journaling through reminders and motivational prompts.

The MindVault – AI Journaling App therefore bridges the gap between emotional expression and technological intelligence, turning journaling into a meaningful, secure, and reflective digital experience.

F. Architecture Design

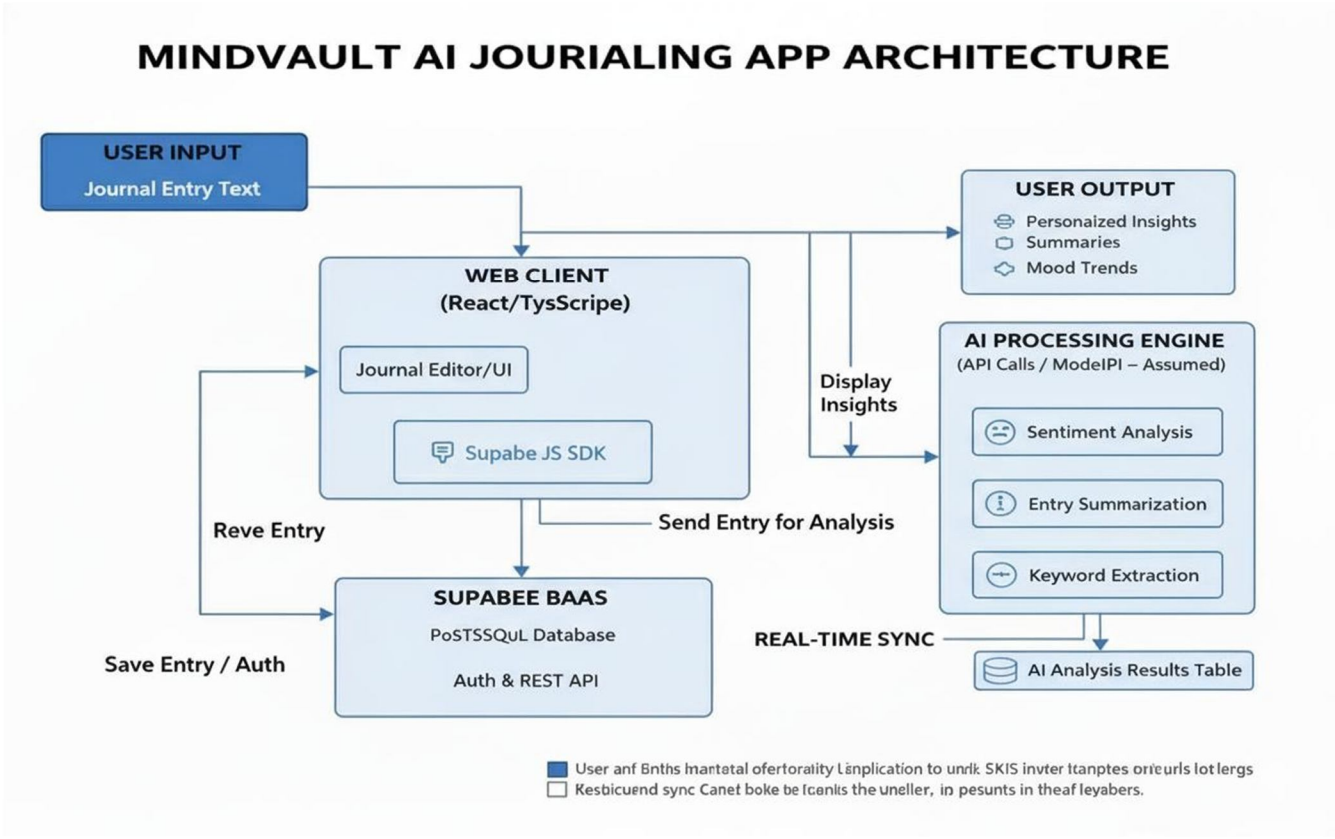


Figure 3.1: Architecture Diagram

1) User Interface (UI) Layer (Web Client - React/TypeScript)

- Purpose: Provides the primary interactive experience for the user, allowing them to create, manage, and view journal entries, and interact with AI-generated insights.
- Key Components:
  - Journal Editor/UI: Renders the application's visual elements, handles user input for journal entries, displays lists of entries, and presents AI-driven insights.
- Technologies: React, TypeScript, Vite, Tailwind CSS.

2) Data Access & Authentication Layer (Supabase BaaS)

- Purpose: Securely manages user authentication, stores all journal entry data, and provides a robust, scalable backend API for the frontend application.
- Key Components:
  - PostgreSQL Database: The primary data store for all user accounts, journal entries, and associated metadata (e.g., timestamps, AI analysis results). Supabase manages this fully.
  - Authentication Service: Handles user registration, login, session management, and authorization using industry-standard protocols.
  - RESTful API: Automatically generated by Supabase from the database schema, allowing the client application to perform CRUD (Create, Read, Update, Delete) operations on journal entries.
  - Supabase JS SDK: A client-side library used by the Web Client to interact with Supabase services (Auth, Database API, Realtime).
- Technologies: Supabase (PostgreSQL, Auth, API Gateway).

3) AI Processing Layer (AI Processing Engine - Assumed: OpenAI API)

- Purpose: Processes raw journal entry text to extract meaningful insights and generate additional content, enhancing the journaling experience.
- Key Components (Assumed functionalities):
  - Sentiment Analysis Module: Analyzes the emotional tone of a journal entry (e.g., positive, negative, neutral) to help users track their mood over time.
  - Entry Summarization Module: Generates concise summaries of longer journal entries, allowing for quick review or identifying key takeaways.
  - Keyword Extraction Module: Identifies important keywords and themes within entries, aiding in categorizing and searching journal content.
  - AI Service API (e.g., OpenAI API): The external interface to the underlying AI models that perform the actual analysis. The Web Client sends text to this API for processing.
- Technologies: OpenAI API (or other chosen LLM provider), potentially Supabase Edge Functions if a proxy layer is needed for API key security or rate limiting.

4) Data Synchronization & Insights Layer (Real-time Sync & User Output)

- Purpose: Stores the results of AI processing and presents these insights back to the user in an actionable format.
- Key Components:
  - AI Analysis Results Table: A dedicated table within the PostgreSQL database (managed by Supabase) to store the output from the AI Processing Layer, linked to specific journal entries.
  - User Output/Insights Display: The UI components in the Web Client responsible for visually representing the AI-generated "Personalized Insights," "Summaries," and "Mood Trends" to the user.
- Technologies: Supabase PostgreSQL, Supabase Realtime (optional), React/TypeScript for rendering.

G. Data Flow Diagrams

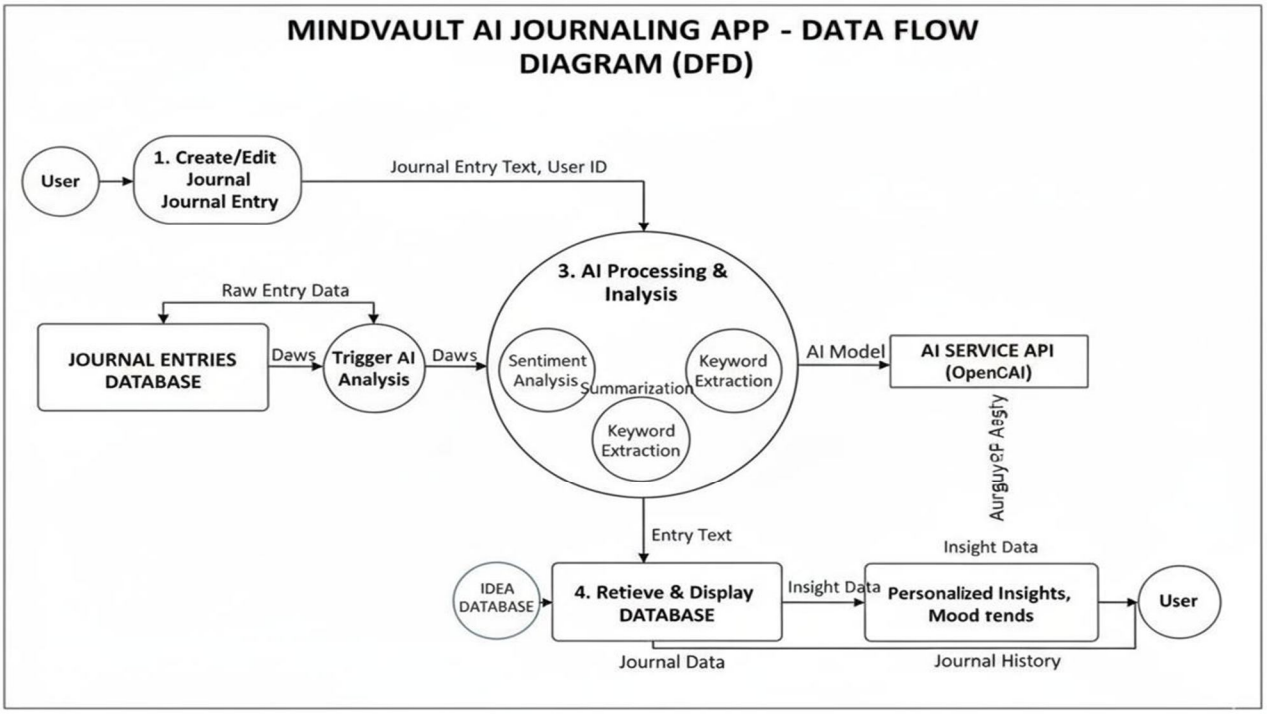


Figure 3.2: Data Flow Diagram

The MindVault AI Journaling App processes user entries through three main operational phases:

1) Input Phase:
- User Action: User creates/edits journal text in the Web Client.
- Data Flow: Entry text & user ID sent to Supabase BaaS and saved in JOURNAL ENTRIES DATABASE.

2) Processing Phase:
- Trigger: New/updated entry triggers AI analysis.
- Data Flow: Entry text sent to AI Service API (OpenAI).
- AI Action: AI performs Sentiment Analysis, Summarization, Keyword Extraction.
- Data Flow: Generated "Insight Data" saved in AI INSIGHTS DATABASE.

3) Output Phase:
- User Action: User views journal history or specific entry.
- Data Flow: Web Client retrieves original entry (from JOURNAL ENTRIES DATABASE) and AI insights (from AI INSIGHTS DATABASE).
- Display: Personalized Insights, Summaries, and Mood Trends are displayed to the user.

IV. SYSTEM REQUIREMENT

A. Hardware Requirements

1) Minimum Requirements
- Processor: Intel Core i3 or above
- RAM: 4 GB (8 GB recommended)
- Hard Disk: 250 GB or more
- Monitor: 1024×768 resolution or higher
- Input Devices: Keyboard and Mouse
- Internet: Required for API access

2) Recommended Requirements:
- Processor: Intel Core i5 or higher
- RAM: 8 GB or more
- Storage: 500 GB SSD

Network: Stable internet connection

B. Software Requirements

1) Operating System
- Windows 10/11 or Linux (Ubuntu) Programming Language:
- Python 3.10 or above

2) Frameworks and Libraries
- TensorFlow / PyTorch: Used for building and training AI and emotion detection models that analyze journal entries.
- NLTK (Natural Language Toolkit): Helps process and analyze text data for sentiment and mood detection.
- spaCy: Used for text tokenization, keyword extraction, and part-of-speech tagging in natural language processing.
- Flask / Django: Provides a backend framework for handling data requests, API connections, and user management.
- React / Flutter: Used for building a dynamic, responsive, and user-friendly front- end interface.
- Firebase / MongoDB: Offers secure, scalable cloud database solutions for storing journal entries and user data.
- Matplotlib / Plotly: Used for visualizing emotional trends and mood analytics in graphical formats.
- Encryption Libraries (e.g., PyCryptodome): Ensure the protection and confidentiality of personal journal data.

By integrating these frameworks and libraries, **MindVault** ensures a seamless blend of AI intelligence, emotional insight, and secure data management — creating a powerful, modern journaling experience for users.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XI Nov 2025- Available at www.ijraset.com

V. IMPLEMENTATION

A. Data Collection

Data collection plays a vital role in developing the MindVault – AI Journaling App, as it directly impacts the accuracy of mood detection and personalized insights. The app uses emotional and sentiment-based text data to train its AI models for understanding user emotions through journaling entries.

The initial datasets were gathered from public sources such as Kaggle Emotion Dataset and Sentiment140, which include thousands of emotion-labeled text samples. Additionally, anonymous user journaling data and feedback were collected during the testing phase to fine-tune the model's predictions.

Data Sources

1) Public Sentiment Datasets– For training emotion recognition models.
2) User Journal Entries– Used for improving personalization.
3) Feedback Data– Helps enhance AI recommendations.
4) Synthetic Data – Generated to balance emotional categories.

All collected data was preprocessed through tokenization, stop-word removal, and normalization. To protect user privacy, every entry was anonymized and securely stored.

Through effective data collection, MindVault ensures reliable emotional analysis and meaningful AI-powered journaling experiences.

B. Component Design

The component design of the MindVault – AI Journaling App divides the system into smaller, manageable modules, each responsible for specific functions. This modular design ensures better organization, flexibility, and easy maintenance of the system.

The main components of the system include:

1) User Interface Component – Handles all user interactions such as login, writing journal entries, and viewing mood insights.
2) Emotion Analysis Component – Uses AI and NLP techniques to analyze journal text and detect user emotions.
3) Database Component – Stores user information, journal entries, and emotion analysis results securely.
4) Insight Generation Component – Provides personalized recommendations and mood summaries based on detected emotions.
5) Security Component – Ensures user privacy through encryption and secure authentication methods.

These components work together to make MindVault efficient, user-friendly, and capable of providing intelligent emotional insights.

C. Software Description

The MindVault – AI Journaling App is developed using modern software tools and technologies that support artificial intelligence and natural language processing. The software is designed as a web-based application with a user-friendly interface and intelligent backend functionalities.

The frontend of the application is built using HTML, CSS, and JavaScript, ensuring smooth interaction and attractive design. React.js is used to enhance responsiveness and improve the overall user experience.

The backend is developed using Python with the Flask framework, which handles user requests, emotion analysis, and communication with the database. The AI models are implemented using TensorFlow and Natural Language Toolkit (NLTK) for sentiment and mood detection.

The database is managed using MySQL, which stores journal entries, user profiles, and emotion records securely. Additionally, RESTful APIs are used to connect the frontend and backend efficiently.

The software is lightweight, scalable, and compatible with multiple devices, making it accessible for users who wish to track their emotions and mental well-being through daily journaling.

.

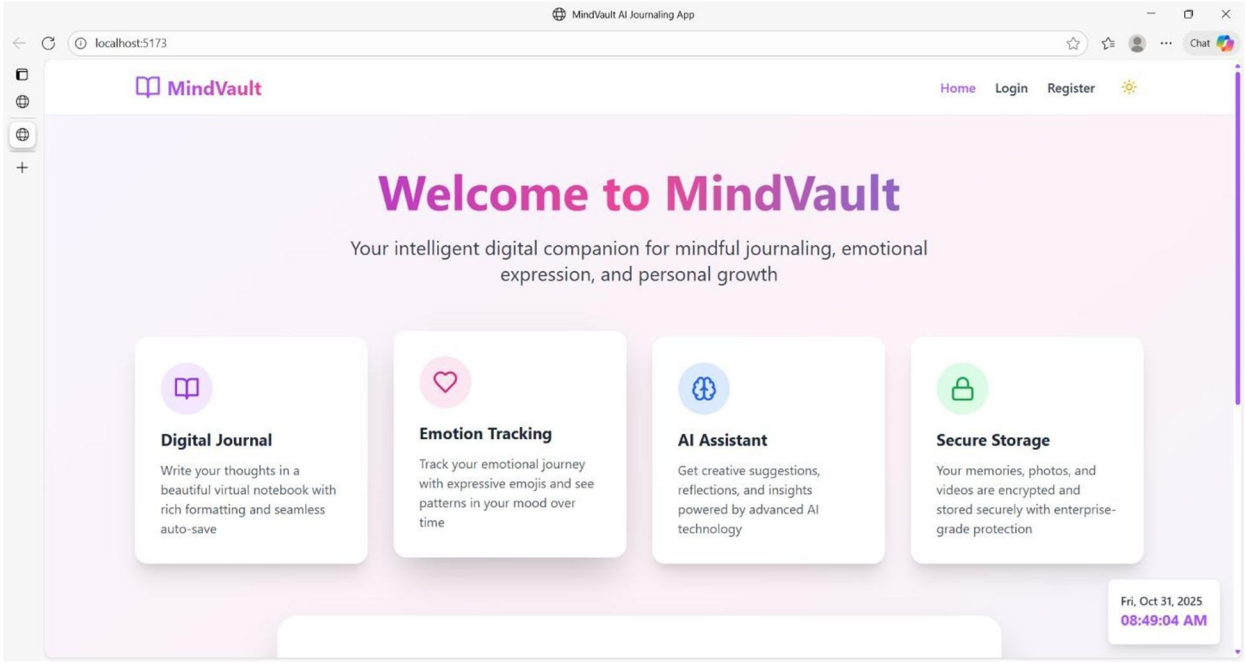
D. Result

The implementation of the MindVault – AI Journaling App successfully achieved its objectives. The system allows users to write daily journal entries and automatically analyzes their emotions using artificial intelligence and natural language processing.

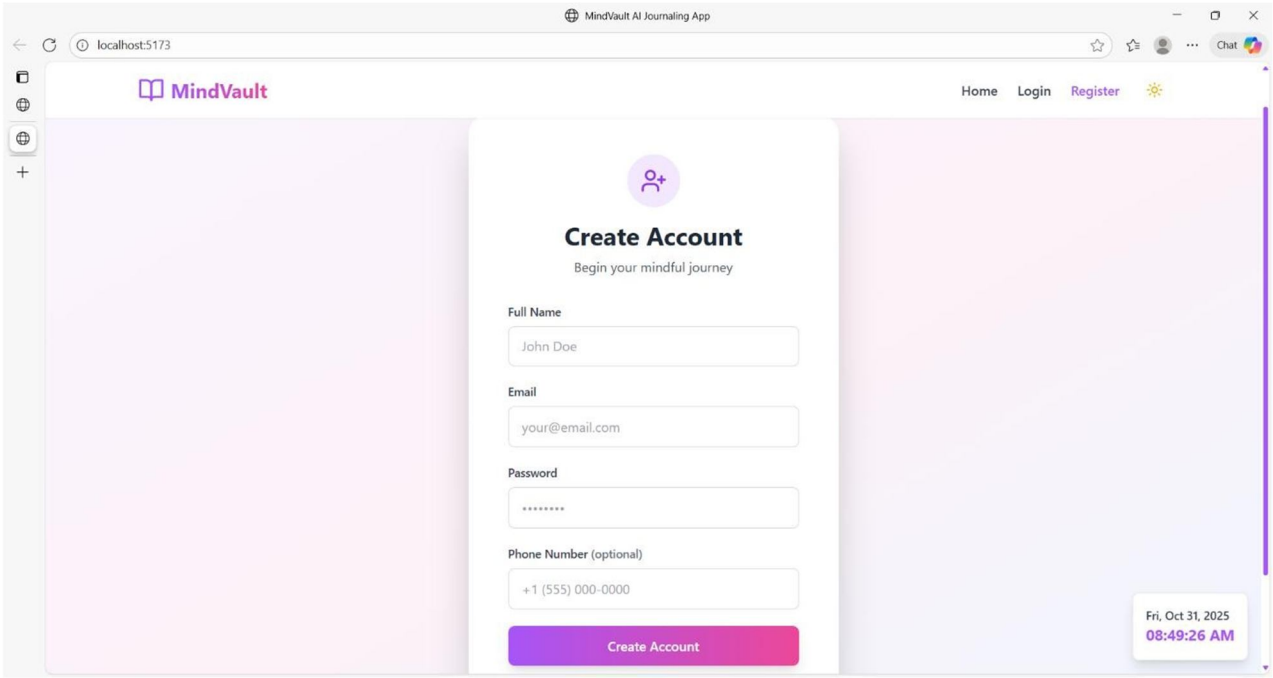
The results show that the system effectively combines journaling with emotional analysis, creating a helpful digital space for self-reflection and emotional awareness. Overall, the app performed efficiently with high accuracy in emotion detection and provided a positive user experience.

The App Flow

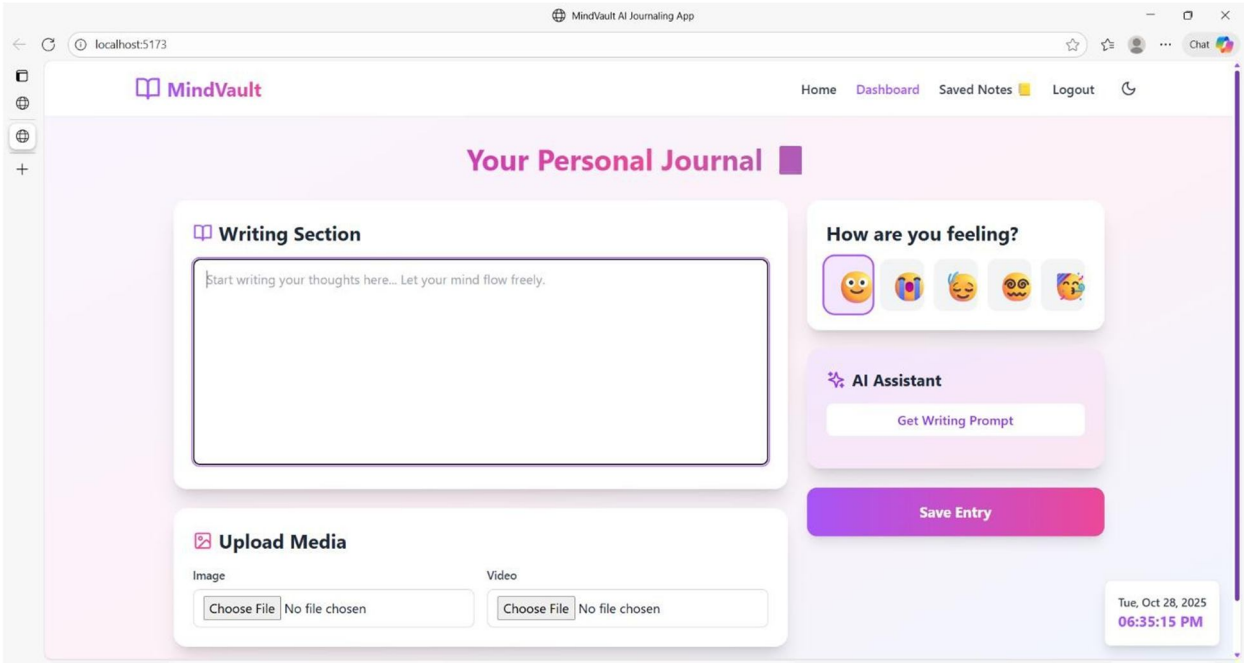
The MindVault – AI Journaling App follows a simple and user-friendly flow that helps users navigate easily between different sections while maintaining focus on journaling and emotional tracking. Each page in the app serves a specific purpose to ensure a smooth and effective user experience.



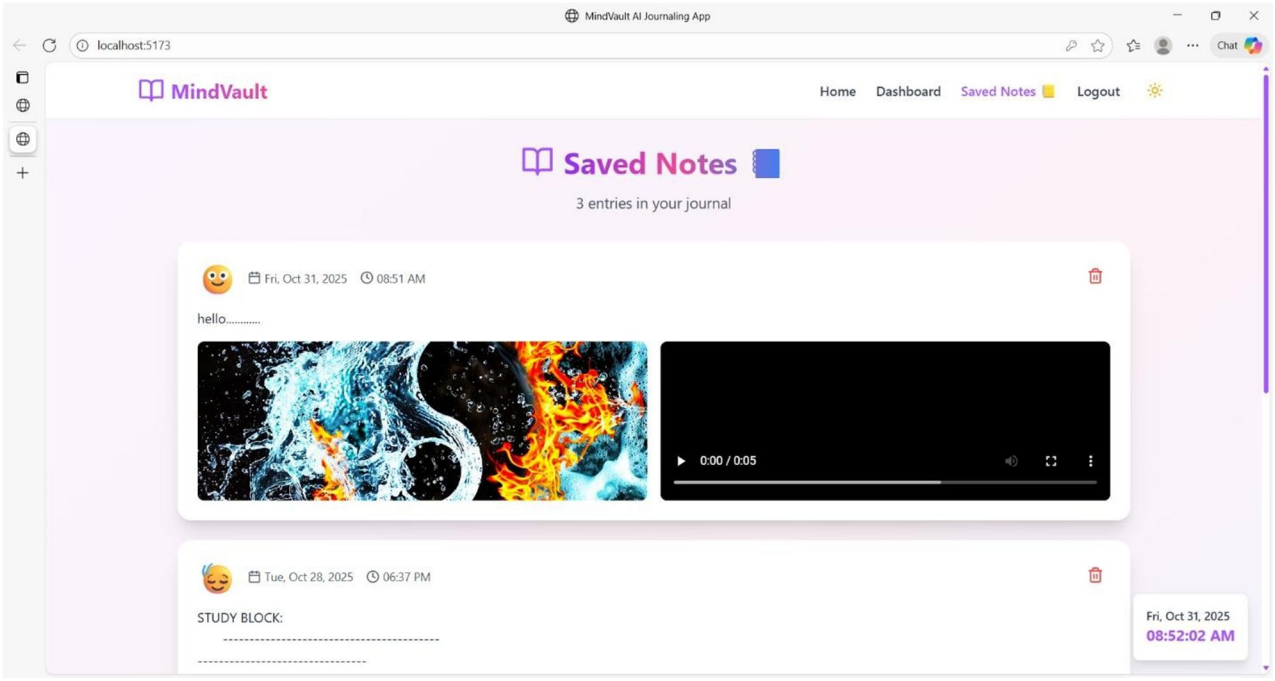
The **Home Page** is the first screen that welcomes the user. It provides an overview of the app and gives quick access to login, registration, and key features. From here, users can start journaling or access their saved notes once logged in.



The **Login/Register Page** allows users to securely create an account or log in to an existing one. This ensures that each user's data, journals, and mood analysis are stored privately and can be accessed anytime.



The **Dashboard Page** is the main workspace where users can write their daily journal entries. Once an entry is submitted, the AI analyzes it using natural language processing to detect emotions such as happiness, sadness, calmness, or stress. The analyzed results are displayed as emotion insights and visual graphs.



The **Saved Notes Page** displays all the journal entries written by the user in one place. Users can view, edit, or delete past entries. It helps them reflect on their thoughts over time and track emotional changes based on past journals.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XI Nov 2025- Available at www.ijraset.com

VI. CONCLUSION AND FUTURE ENHANCEMENT

A. *Conclusion*

The MindVault – AI Journaling App successfully integrates artificial intelligence with digital journaling to support emotional well-being and self-reflection. By analyzing users' written entries, the app identifies emotions, tracks mood changes, and provides meaningful insights and suggestions.

This project demonstrates how technology can be used to improve mental health awareness through regular journaling and emotional analysis. The AI-driven approach ensures accuracy, privacy, and personalization, making the system useful for individuals seeking emotional balance in their daily lives.

Overall, the app achieved its intended goals of combining journaling with intelligent mood tracking, offering users a reliable and user-friendly digital companion for personal growth and mindfulness.

In conclusion, MindVault stands as a powerful AI-based journaling platform that not only encourages emotional expression but also promotes self-awareness and mental wellness. It successfully fulfills its objectives by merging technology with human emotion, offering users a modern, intelligent, and therapeutic journaling experience. With further development and continuous AI improvement, MindVault has the potential to become an essential tool for mental health support in the digital era.

B. *Future Scope*

The MindVault – AI Journaling App has vast potential for future enhancement and expansion. While the current version provides efficient emotion detection, journaling support, and insightful analytics, several improvements can be made to make the system more advanced, intelligent, and accessible to a wider audience.

In the future, the application can be developed into a mobile platform for both Android and iOS devices, allowing users to conveniently record their emotions anytime and anywhere. Integration with voice journaling and speech emotion recognition would make the app more interactive, enabling users to express feelings verbally rather than through text.

Another major improvement could involve the use of deep learning models such as BERT or GPT-based architectures to increase the accuracy of emotional analysis and to understand complex human sentiments better. The system could also include context-based emotional prediction, where the AI learns user behavior patterns and provides proactive suggestions for mental well-being.

Additionally, future versions of MindVault can integrate cloud storage and synchronization, allowing users to access their data securely from multiple devices. Advanced data visualization dashboards can help display long-term mood trends more clearly, giving users a better understanding of their emotional progress over time.

The app could also be extended to include community support features, connecting users with similar emotional patterns or mental health professionals when necessary. This would transform MindVault into not only a journaling app but also a comprehensive emotional wellness platform.

Appendices Source Code

Dashboard.tsx

```
import { useState } from 'react';
import { supabase } from '../lib/supabase';
import { useAuth } from '../contexts/AuthContext';
import { BookOpen, Image, Video, Sparkles } from 'lucide-react'; const EMOTIONS = [
  { emoji: '□', label: 'Happy' },
  { emoji: '□', label: 'Sad' },
  { emoji: '□↕□', label: 'Neutral' },
  { emoji: '□□', label: 'Confused' },
  { emoji: '□', label: 'Excited' },
];
interface DashboardProps {
  onNavigate: (page: string) => void;
}
export const Dashboard = ({ onNavigate }: DashboardProps) => { const { user } = useAuth();
  const [content, setContent] = useState('');
  const [selectedEmotion, setSelectedEmotion] = useState('□'); const [imageFile, setImageFile] = useState<File | null>(null); const
```

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XI Nov 2025- Available at www.ijraset.com

```
[videoFile, setVideoFile] = useState<File | null>(null); const [imagePreview, setImagePreview] = useState<string>(''); const
[videoPreview, setVideoPreview] = useState<string>(''); const [aiSuggestion, setAiSuggestion] = useState('');
const [loading, setLoading] = useState(false); const [message, setMessage] = useState('');
const handleImageChange = (e: React.ChangeEvent<HTMLInputElement>) => { const file = e.target.files?.[0];
  if (file) {
    setImageFile(file);
    const reader = new FileReader(); reader.onloadend = () => {
      setImagePreview(reader.result as string);
    };
    reader.readAsDataURL(file);
  }
};
const handleVideoChange = (e: React.ChangeEvent<HTMLInputElement>) => { const file = e.target.files?.[0];
  if (file) {
    setVideoFile(file);
    const reader = new FileReader(); reader.onloadend = () => {
      setVideoPreview(reader.result as string);
    };
    reader.readAsDataURL(file);
  }
};
const uploadFile = async (file: File, bucket: string) => {
  const fileExt = file.name.split('.').pop();
  const fileName = `${user?.id}-${Date.now()}.${fileExt}`; const filePath = `${fileName}`;
  const { error } = await supabase.storage.from(bucket).upload(filePath, file); if (error) throw error;
  const { data: { publicUrl } } = supabase.storage.from(bucket).getPublicUrl(filePath); return publicUrl;
};
const generateAiSuggestion = () => { const suggestions = [
    "What brought you joy today? Reflect on the small moments that made you smile.", "Consider writing about a challenge you
    faced and how you overcame it.", "Describe your ideal day. What would you do? Who would you spend it with?", "Think about
    three things you're grateful for right now.",
    "Write a letter to your future self. What advice would you give?", "Reflect on a person who has positively influenced your life.",
    "What emotions are you feeling right now? Explore them without judgment.", "Describe a memory that always makes you feel
    peaceful.",
  ];
  const random = suggestions[Math.floor(Math.random() * suggestions.length)]; setAiSuggestion(random);
};
const handleSave = async () => { if (!content.trim()) {
    setMessage('Please write something before saving'); return;
  }
  setLoading(true); setMessage('');

  try {
    let imageUrl = null; let videoUrl = null; if (imageFile) {
      imageUrl = await uploadFile(imageFile, 'images');
    }
    if (videoFile) {
      videoUrl = await uploadFile(videoFile, 'videos');
    }
    const { error } = await supabase.from('notes').insert({ user_id: user?.id,
```



```
        content,
        emotion: selectedEmotion, image_url: imageUrl, video_url: videoUrl,
      });
      if (error) throw error;
      setMessage('Note saved successfully!'); setContent('');
      setImageFile(null); setVideoFile(null);
      setImagePreview('');
        onNavigate('notes');
      }, 1500);
    } catch (err: any) {
      setMessage(err.message || 'Failed to save note');
    } finally {
      setLoading(false);
    }
  };
  return (
    <div className="min-h-screen bg-gradient-to-br from-purple-50 via-pink-50 to-blue-50
dark:from-gray-900 dark:via-gray-800 dark:to-gray-900 transition-colors duration-300 py-8 px- 4">
      <div className="container mx-auto max-w-6xl">
        <h1 className="text-4xl font-bold text-center mb-8 bg-gradient-to-r from-purple-600 via- pink-500 to-blue-500 bg-clip-text
text-transparent">
          Your Personal Journal □
        </h1>
        {message && (
          <div className={`mb-6 p-4 rounded-lg text-center font-semibold ${ message.includes('success')
              ? 'bg-green-100 dark:bg-green-900/30 text-green-700 dark:text-green-400 border border-green-200 dark:border-green-800'
              : 'bg-red-100 dark:bg-red-900/30 text-red-700 dark:text-red-400 border border-red-200 dark:border-red-800'
          }`}>
            {message}
          </div>
        )}

        <div className="grid lg:grid-cols-3 gap-6">
          <div className="lg:col-span-2 space-y-6">
            <div className="bg-white dark:bg-gray-800 rounded-2xl shadow-xl p-6">
              <div className="flex items-center space-x-2 mb-4">
                <BookOpen className="w-6 h-6 text-purple-500" />
                <h2 className="text-2xl font-bold text-gray-800 dark:text-white">Writing Section</h2>
              </div>
              <textarea
                value={content}
                onChange={(e) => setContent(e.target.value)}
                placeholder="Start writing your thoughts here... Let your mind flow freely." className="w-full h-64 px-4 py-3 rounded-
                lg border border-gray-300 dark:border-gray-
600 bg-white dark:bg-gray-700 text-gray-900 dark:text-white focus:ring-2 focus:ring-purple-500 focus:border-transparent transition-
all resize-none"
              />
            </div>
            <div className="bg-white dark:bg-gray-800 rounded-2xl shadow-xl p-6">
              <div className="flex items-center space-x-2 mb-4">
```

```
            <Image className="w-6 h-6 text-pink-500" />
            <h2 className="text-2xl font-bold text-gray-800 dark:text-white">Upload Media</h2>
          </div>
          <div className="grid md:grid-cols-2 gap-4">
            <div>
              <label className="block text-sm font-medium text-gray-700 dark:text-gray-300 mb-
2">
                Image
              </label>
              <input
                type="file"
                accept="image/*" onChange={handleImageChange}
                className="w-full px-3 py-2 border border-gray-300 dark:border-gray-600 rounded-lg bg-white dark:bg-gray-700 text-
gray-900 dark:text-white"
              </label>
              <input
                type="file"
                accept="video/*" onChange={handleVideoChange}
                className="w-full px-3 py-2 border border-gray-300 dark:border-gray-600 rounded-lg bg-white dark:bg-gray-700 text-
gray-900 dark:text-white"
              />
              {videoPreview && (
                <video
                  src={videoPreview} controls
                  className="mt-3 w-full h-32 rounded-lg"
                />
              )}
            </div>
          </div>
        </div>
      </div>
      <div  className="space-y-6">
        <div className="bg-white dark:bg-gray-800 rounded-2xl shadow-xl p-6">
          <h2 className="text-2xl font-bold text-gray-800 dark:text-white mb-4"> How are you feeling?
          </h2>
          <div className="grid grid-cols-5 gap-3">
            {EMOTIONS.map((emotion) => (
              <button key={emotion.emoji}
                onClick={() => setSelectedEmotion(emotion.emoji)}
                className={`text-4xl p-3 rounded-xl transition-all transform hover:scale-110 ${ selectedEmotion === emotion.emoji
                    ? 'bg-purple-100 dark:bg-purple-900 ring-2 ring-purple-500 scale-110'
                    : 'bg-gray-100 dark:bg-gray-700 hover:bg-gray-200 dark:hover:bg-gray-600'
                }`} title={emotion.label}
              >
                {emotion.emoji}
              </button>
            ))}
          </div>
        </div>
        <div className="bg-gradient-to-br from-purple-100 to-pink-100 dark:from-purple-900/50 dark:to-pink-900/50 rounded-2xl
```

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XI Nov 2025- Available at www.ijraset.com

```
shadow-xl p-6">
            <div className="flex items-center space-x-2 mb-4">
              <Sparkles className="w-6 h-6 text-purple-600 dark:text-purple-400" />
              <h2 className="text-xl font-bold text-gray-800 dark:text-white">AI Assistant</h2>
            </div>
            <button
              onClick={generateAiSuggestion}
              className="w-full bg-white dark:bg-gray-800 text-purple-600 dark:text-purple-400 py-2 px-4 rounded-lg font-semibold
mb-4 hover:shadow-md transition-all"
            >
              Get Writing Prompt
            </button>
            {aiSuggestion && (
              <p className="text-gray-700 dark:text-gray-300 italic leading-relaxed">
                {aiSuggestion}
              </p>
            )}
          </div>
          <button
            onClick={handleSave} disabled={loading}
            className="w-full bg-gradient-to-r from-purple-500 to-pink-500 text-white py-4 rounded-xl font-bold text-lg shadow-lg
hover:shadow-xl transform hover:scale-105 transition-all duration-300 disabled:opacity-50 disabled:cursor-not-allowed"
          >
            {loading ? 'Saving...' : 'Save Entry'}
          </button>
        </div>
      </div>
    </div>
  </div>
 );
};
```

Home.tsx

```
import { BookOpen, Brain, Heart, Lock } from 'lucide-react'; interface HomeProps {
 onNavigate: (page: string) => void;
}
        </h1>
        <p className="text-xl md:text-2xl text-gray-700 dark:text-gray-300 max-w-3xl mx-auto leading-relaxed">
          Your intelligent digital companion for mindful journaling, emotional expression, and personal growth
        </p>
      </div>
      <div className="grid md:grid-cols-2 lg:grid-cols-4 gap-8 mb-16">
        <div className="bg-white dark:bg-gray-800 rounded-2xl p-8 shadow-xl hover:shadow-2xl transform hover:-translate-y-2
transition-all duration-300">
          <div className="bg-purple-100 dark:bg-purple-900 w-16 h-16 rounded-full flex items- center justify-center mb-4">
            <BookOpen className="w-8 h-8 text-purple-600 dark:text-purple-400" />
          </div>
          <h3 className="text-xl font-bold mb-3 text-gray-800 dark:text-white">Digital Journal</h3>
          <p className="text-gray-600 dark:text-gray-400">
            Write your thoughts in a beautiful virtual notebook with rich formatting and seamless auto-save
```

```
          </p>
        </div>
        <div className="bg-white dark:bg-gray-800 rounded-2xl p-8 shadow-xl hover:shadow-2xl transform hover:-translate-y-2
transition-all duration-300">
          <div className="bg-pink-100 dark:bg-pink-900 w-16 h-16 rounded-full flex items-center justify-center mb-4">
            <Heart className="w-8 h-8 text-pink-600 dark:text-pink-400" />
          </div>
          <h3 className="text-xl font-bold mb-3 text-gray-800 dark:text-white">Emotion Tracking</h3>
          <p className="text-gray-600 dark:text-gray-400">
            Track your emotional journey with expressive emojis and see patterns in your mood over
time
          </p>
        </div>
        <div className="bg-white dark:bg-gray-800 rounded-2xl p-8 shadow-xl hover:shadow-2xl transform hover:-translate-y-2
transition-all duration-300">
          <div className="bg-blue-100 dark:bg-blue-900 w-16 h-16 rounded-full flex items-center justify-center mb-4">
            <Brain className="w-8 h-8 text-blue-600 dark:text-blue-400" />
          </div>
          <h3 className="text-xl font-bold mb-3 text-gray-800 dark:text-white">AI Assistant</h3>
          <p className="text-gray-600 dark:text-gray-400">
            Get creative suggestions, reflections, and insights powered by advanced AI technology
          </p>
        </div>
        <div className="bg-white dark:bg-gray-800 rounded-2xl p-8 shadow-xl hover:shadow-2xl transform hover:-translate-y-2
transition-all duration-300">
          <div className="bg-green-100 dark:bg-green-900 w-16 h-16 rounded-full flex items- center justify-center mb-4">
            <Lock className="w-8 h-8 text-green-600 dark:text-green-400" />
          </div>
          <h3 className="text-xl font-bold mb-3 text-gray-800 dark:text-white">Secure Storage</h3>
          <p className="text-gray-600 dark:text-gray-400">
            Your memories, photos, and videos are encrypted and stored securely with enterprise- grade protection
          </p>
        </div>
      </div>
      <div className="bg-white dark:bg-gray-800 rounded-3xl p-12 shadow-2xl max-w-4xl mx- auto">
        <h2 className="text-3xl font-bold mb-6 text-gray-800 dark:text-white text-center"> Why Choose MindVault?
        </h2>
        <div className="space-y-4 text-gray-700 dark:text-gray-300 text-lg">
          <p>
            <strong className="text-purple-600 dark:text-purple-400">Holistic Expression:</strong> Combine text, images, and videos
to capture the full richness of your experiences and memories.
          </p>
          <p>
            <strong className="text-pink-600 dark:text-pink-400">Emotional Intelligence:</strong> Our emotion tracker helps you
understand your feelings and identify patterns in your mental wellness journey.
          </p>
          <p>
            <strong className="text-blue-600 dark:text-blue-400">AI-Powered Insights:</strong> Receive thoughtful prompts,
creative suggestions, and personalized reflections to deepen your self-awareness.
          </p>
```

```
            <p>
              <strong className="text-green-600 dark:text-green-400">Privacy First:</strong> Your journal is completely private. We use advanced encryption to ensure your thoughts remain yours alone.
            </p>
          </div>
          <div className="mt-8 text-center">
            <button
              onClick={() => onNavigate('register')}
              className="bg-gradient-to-r from-purple-500 to-pink-500 text-white px-8 py-4 rounded- full font-bold text-lg shadow-lg hover:shadow-xl transform hover:scale-105 transition-all duration-300"
            >
              Start Your Journey Today
            </button>
          </div>
        </div>
      </div>
    </div>
  );
};
```

Login.tsx

```
import { useState } from 'react';
import { useAuth } from '../contexts/AuthContext'; import { LogIn } from 'lucide-react';
interface LoginProps {
  onNavigate: (page: string) => void;
}
export const Login = ({ onNavigate }: LoginProps) => { const [email, setEmail] = useState('');
  const [password, setPassword] = useState(''); const [error, setError] = useState('');
  const [loading, setLoading] = useState(false); const { signIn } = useAuth();
  const handleSubmit = async (e: React.FormEvent) => { e.preventDefault();
    setError('');
    setLoading(true); try {
      await signIn(email, password); onNavigate('dashboard');
    } catch (err: any) {
      setError(err.message || 'Failed to sign in');
    } finally {
      setLoading(false);
    }
  };
  return (
    <div className="min-h-screen bg-gradient-to-br from-purple-50 via-pink-50 to-blue-50 dark:from-gray-900 dark:via-gray-800 dark:to-gray-900 transition-colors duration-300 flex items- center justify-center px-4">
      <div className="bg-white dark:bg-gray-800 rounded-3xl shadow-2xl p-8 md:p-12 w-full max- w-md">
        <div className="text-center mb-8">
          <div className="bg-purple-100 dark:bg-purple-900 w-16 h-16 rounded-full flex items- center justify-center mx-auto mb-4">
            <LogIn className="w-8 h-8 text-purple-600 dark:text-purple-400" />
          </div>
          <h2 className="text-3xl font-bold text-gray-800 dark:text-white mb-2">Welcome Back</h2>
          <p className="text-gray-600 dark:text-gray-400">Sign in to continue your journey</p>
        </div>
        {error && (
```

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XI Nov 2025- Available at www.ijraset.com

```
        <div className="bg-red-50 dark:bg-red-900/30 border border-red-200 dark:border-red- 800 text-red-700 dark:text-red-400
px-4 py-3 rounded-lg mb-6">
          {error}
        </div>
      )}
      <form onSubmit={handleSubmit} className="space-y-6">
        <div>
          <label className="block text-sm font-medium text-gray-700 dark:text-gray-300 mb-2"> Email
          </label>
          <input
            type="email" value={email}
            onChange={(e) => setEmail(e.target.value)} required
            className="w-full px-4 py-3 rounded-lg border border-gray-300 dark:border-gray-600 bg-white dark:bg-gray-700 text-
gray-900 dark:text-white focus:ring-2 focus:ring-purple-500 focus:border-transparent transition-all"
            placeholder="your@email.com"
          />
        </div>
        <div>
          <label className="block text-sm font-medium text-gray-700 dark:text-gray-300 mb-2"> Password
          </label>
          <input
            type="password" value={password}
            onChange={(e) => setPassword(e.target.value)} required
            className="w-full px-4 py-3 rounded-lg border border-gray-300 dark:border-gray-600 bg-white dark:bg-gray-700 text-
gray-900 dark:text-white focus:ring-2 focus:ring-purple-500 focus:border-transparent transition-all"
            placeholder="••••••••"
          />
        </div>
        <button
          type="submit" disabled={loading}
          className="w-full bg-gradient-to-r from-purple-500 to-pink-500 text-white py-3 rounded- lg font-semibold shadow-lg
hover:shadow-xl transform hover:scale-105 transition-all duration-
300 disabled:opacity-50 disabled:cursor-not-allowed"
        >
          {loading ? 'Signing in...' : 'Sign In'}
        </button>
      </form>

  <p className="mt-6 text-center text-gray-600 dark:text-gray-400"> Don't have an account?{' '}
        <button
          onClick={() => onNavigate('register')}
          className="text-purple-600 dark:text-purple-400 font-semibold hover:underline"
        >
          Register here
        </button>
      </p>
    </div>
  </div>
 );
};
```


Register.tsx

```
import { useState } from 'react';
import { useAuth } from '../contexts/AuthContext'; import { UserPlus } from 'lucide-react';
interface RegisterProps {
  onNavigate: (page: string) => void;
}
export const Register = ({ onNavigate }: RegisterProps) => { const [name, setName] = useState('');
  const [email, setEmail] = useState('');
  const [password, setPassword] = useState(''); const [phone, setPhone] = useState('');
  const [error, setError] = useState('');

  const [loading, setLoading] = useState(false); const { signUp } = useAuth();
  const handleSubmit = async (e: React.FormEvent) => { e.preventDefault();
    setError('');
    setLoading(true); try {
  return (
    <div className="min-h-screen bg-gradient-to-br from-purple-50 via-pink-50 to-blue-50
dark:from-gray-900 dark:via-gray-800 dark:to-gray-900 transition-colors duration-300 flex items- center justify-center px-4 py-12">
      <div className="bg-white dark:bg-gray-800 rounded-3xl shadow-2xl p-8 md:p-12 w-full max- w-md">
        <div className="text-center mb-8">
          <div className="bg-purple-100 dark:bg-purple-900 w-16 h-16 rounded-full flex items- center justify-center mx-auto mb-4">
            <UserPlus className="w-8 h-8 text-purple-600 dark:text-purple-400" />
          </div>
          <h2 className="text-3xl font-bold text-gray-800 dark:text-white mb-2">Create Account</h2>
          <p className="text-gray-600 dark:text-gray-400">Begin your mindful journey</p>
        </div>
        {error && (
          <div className="bg-red-50 dark:bg-red-900/30 border border-red-200 dark:border-red- 800 text-red-700 dark:text-red-400
px-4 py-3 rounded-lg mb-6">
            {error}
          </div>
        )}

        <form onSubmit={handleSubmit} className="space-y-5">
          <div>
            <label className="block text-sm font-medium text-gray-700 dark:text-gray-300 mb-2"> Full Name
            </label>
            <input
              type="text" value={name}
              onChange={(e) => setName(e.target.value)} required
            />
          </div>
          <div>
            <label className="block text-sm font-medium text-gray-700 dark:text-gray-300 mb-2"> Email
            </label>
            <input
              type="email" value={email}
              onChange={(e) => setEmail(e.target.value)} required
              className="w-full px-4 py-3 rounded-lg border border-gray-300 dark:border-gray-600 bg-white dark:bg-gray-700 text-
gray-900 dark:text-white focus:ring-2 focus:ring-purple-500 focus:border-transparent transition-all"
```

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XI Nov 2025- Available at www.ijraset.com

```
            placeholder="your@email.com"
           />
          </div>

          <div>
           <label className="block text-sm font-medium text-gray-700 dark:text-gray-300 mb-2"> Password
           </label>
           <input
            type="password" value={password}
            onChange={(e) => setPassword(e.target.value)} required
            minLength={6}
            className="w-full px-4 py-3 rounded-lg border border-gray-300 dark:border-gray-600 bg-white dark:bg-gray-700 text-
gray-900 dark:text-white focus:ring-2 focus:ring-purple-500 focus:border-transparent transition-all"
            placeholder="••••••••"
           />
          </div>
          <div>
           <label className="block text-sm font-medium text-gray-700 dark:text-gray-300 mb-2"> Phone Number <span
            className="text-gray-500">(optional)</span>
           </label>
           <input
            type="tel" value={phone}
            onChange={(e) => setPhone(e.target.value)}
            className="w-full px-4 py-3 rounded-lg border border-gray-300 dark:border-gray-600 bg-white dark:bg-gray-700 text-
gray-900 dark:text-white focus:ring-2 focus:ring-purple-500 focus:border-transparent transition-all"
            placeholder="+1 (555) 000-0000"

           />
          </div>
          <button
           type="submit" disabled={loading}
           className="w-full bg-gradient-to-r from-purple-500 to-pink-500 text-white py-3 rounded- lg font-semibold shadow-lg
hover:shadow-xl transform hover:scale-105 transition-all duration-
300 disabled:opacity-50 disabled:cursor-not-allowed"
          >
           {loading ? 'Creating account...' : 'Create Account'}
          </button>
        </form>
        <p className="mt-6 text-center text-gray-600 dark:text-gray-400"> Already have an account?{' '}
          <button
           onClick={() => onNavigate('login')}
           className="text-purple-600 dark:text-purple-400 font-semibold hover:underline"
          >
           Sign in here
          </button>
        </p>
       </div>
      </div>
    );
};
```


ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XI Nov 2025- Available at www.ijraset.com

App.tsx

```
import { useState, useEffect } from 'react';
import { AuthProvider, useAuth } from './contexts/AuthContext'; import { ThemeProvider } from './contexts/ThemeContext'; import
{ Header } from './components/Header';
import { DateTimeDisplay } from './components/DateTimeDisplay'; import { Home } from './pages/Home';
import { Login } from './pages/Login'; import { Register } from './pages/Register';
import { Dashboard } from './pages/Dashboard'; import { SavedNotes } from './pages/SavedNotes';
type Page = 'home' | 'login' | 'register' | 'dashboard' | 'notes'; const AppContent = () => {
  const { user, loading } = useAuth();
  const [currentPage, setCurrentPage] = useState<Page>('home'); useEffect(() => {
    if (!loading) {
      if (user && currentPage === 'home') { setCurrentPage('dashboard');
      } else if (!user && (currentPage === 'dashboard' || currentPage === 'notes')) { setCurrentPage('home');
      }
  if (loading) { return (
      <div className="min-h-screen bg-gradient-to-br from-purple-50 via-pink-50 to-blue-50 dark:from-gray-900 dark:via-gray-800
dark:to-gray-900 flex items-center justify-center">
        <div className="text-2xl font-semibold text-gray-700 dark:text-gray-300">Loading...</div>
      </div>

    );
  }
  return (
    <div className="min-h-screen">
      <Header currentPage={currentPage} onNavigate={handleNavigate} />
      {currentPage === 'home' && <Home onNavigate={handleNavigate} />}
      {currentPage === 'login' && <Login onNavigate={handleNavigate} />}
      <DateTimeDisplay />
    </div>
  );
};
function App() { return (
    <ThemeProvider>
      <AuthProvider>
                                                                                    <AppContent />
                                                                                  </AuthProvider>
                                                                                </ThemeProvider>

  );
}
```

Index.css

```
@tailwind base;
@tailwind components; @tailwind utilities; @layer base {
  body {
    @apply transition-colors duration-300;
  }
  * {
    @apply scrollbar-thin scrollbar-thumb-purple-500 scrollbar-track-gray-200 dark:scrollbar-track- gray-800;
  }
}
@layer utilities {
```



```
  .animate-fade-in {
    animation: fadeIn 0.8s ease-in;
  }
  @keyframes fadeIn { from {
      opacity: 0;
      transform: translateY(10px);
    }
    to {
      opacity: 1;
      transform: translateY(0);
    }
  }
  .scrollbar-thin {
    scrollbar-width: thin;
  }
  .scrollbar-thumb-purple-500 {
    scrollbar-color: #a855f7 transparent;
  }
  .scrollbar-thumb-purple-500::-webkit-scrollbar-thumb { background-color: #a855f7;
    border-radius: 4px;
  }
  .scrollbar-track-gray-200::-webkit-scrollbar-track { background-color: #e5e7eb;
  }
  .dark .scrollbar-track-gray-800::-webkit-scrollbar-track {
  }
}
```

Main.tsx

```
import { StrictMode } from 'react';
import { createRoot } from 'react-dom/client'; import App from './App.tsx';
import './index.css';


createRoot(document.getElementById('root')!).render(
  <StrictMode>
    <App />
  </StrictMode>
);
```

VII. ACKNOWLEDGEMENT

It is one of the most efficient tasks in life to choose the appropriate words to express one's gratitude to the beneficiaries. We are very much grateful to God who helped us all the way through the project and how molded us into what we are today.

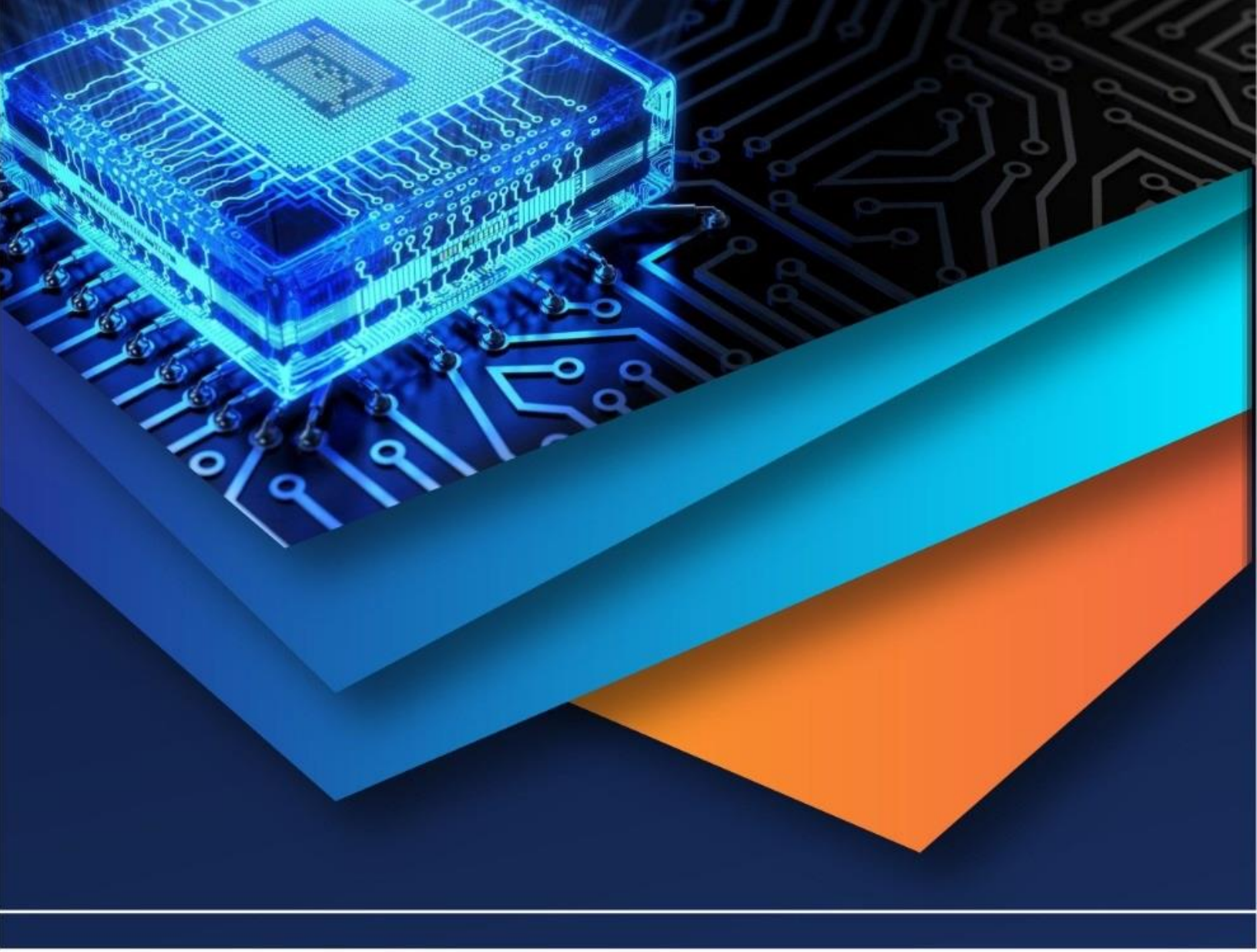
We are grateful to our beloved Principal Dr. R. RADHAKRISHNAN, M.E., Ph.D., Adhiyamaan College of Engineering (Autonomous), Hosur for providing the opportunity to do this work in premises.

We acknowledge our heartful gratitude to Dr. G. FATHIMA, M.E., Ph.D., Professor and Head of the Department, Department of Computer Science and Engineering, Adhiyamaan College of Engineering (Autonomous), Hosur, for her guidance and valuable suggestions and encouragement throughout this project and made us to complete this project successfully.

We are highly indebted to Mrs. K. TAMILSELVI, M.Tech., Supervisor, Assistant Professor, Department of Computer Science and Engineering, Adhiyamaan College of Engineering (Autonomous), Hosur, whose immense support encouragement and valuable guidance were responsible to complete the project successfully. We also extent our thanks to Project Coordinator and all Staff Members for their support in complete this project successfully. Finally, we would like to thank to our parents, without their motivational and support would not have been possible for us to complete this project successfully.

REFERENCES

[1] Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.
[2] Jurafsky, D., & Martin, J. H. (2020). Speech and Language Processing (3rd ed.). Prentice Hall.
[3] Aggarwal, C. C. (2018). Neural Networks and Deep Learning: A Textbook. Springer.
[4] Liu, B. (2015). Sentiment Analysis: Mining Opinions, Sentiments, and Emotions. Cambridge University Press.
[5] Han, J., Kamber, M., & Pei, J. (2012). Data Mining: Concepts and Techniques (3rd ed.). Morgan Kaufmann.
[6] Géron, A. (2019). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (2nd ed.). O'Reilly Media.
[7] Patil, A., & Kulkarni, R. (2022). AI Applications in Mental Health and Well- Being. International Journal of Artificial Intelligence Research.
[8] Tan, P. N., Steinbach, M., & Kumar, V. (2019). Introduction to Data Mining. Pearson Education.
[9] Harrington, P. (2018). Machine Learning in Action. Manning Publications.
[10] Manning, C. D., Raghavan, P., & Schütze, H. (2008). Introduction to Information Retrieval. Cambridge University Press.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY