IN PARTNERSHIP WITH PLYMOUTH UNIVERSITY

Name: **Samarasinghe Methmal**
Student Reference Number: **10898561**

Module Code: **PUSL3190**	Module Name: **Computing Project**
Coursework Title: **Interim Report**	
Deadline Date: **12/03/2025**	Member of staff responsible for coursework: **Mr. Gayan Perera**
Programme: **BSc. (Hons) Software Engineering**	

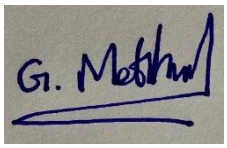
Please note that University Academic Regulations are available under Rules and Regulations on the University website www.plymouth.ac.uk/studenthandbook.

Group work: please list all names of all participants formally associated with this work and state whether the work was undertaken alone or as part of a team. Please note you may be required to identify individual responsibility for component parts.

We confirm that we have read and understood the Plymouth University regulations relating to Assessment Offences and that we are aware of the possible penalties for any breach of these regulations. We confirm that this is the independent work of the group.

Signed on behalf of the group:

Individual assignment: ***I confirm that I have read and understood the Plymouth University regulations relating to Assessment Offences and that I am aware of the possible penalties for any breach of these regulations. I confirm that this is my own independent work.***

Signed: 

Use of translation software: failure to declare that translation software or a similar writing aid has been used will be treated as an assessment offence.

I * have ~~used~~/not used translation software.

If used, please state name of software……

Overall mark _____% **Assessors Initials _____** **Date_____**

PUSL3190 – Computing Project

Interim Report

MoodSync – Mental Health Monitoring System

Mr. Gayan Perera

Samarasinghe Methmal

10898561

BSc. (Hons) Software Engineering

Table of Contents

Table of Figures

1. Introduction

1.1. Introduction

Mental health care has become a more significant issue in the world today, especially in the modern world where daily life pressures, social expectations, and technological advancements have played a huge role in the rising of stress levels, anxiety, and depression. These above-mentioned mental health issues not only affect people's well-being but also have challenged the entire healthcare system. Although there is rising awareness of mental health problems, there remains a significant gap in personalized and accessible mental health care solutions. People frequently hesitate to get help due to the cultural stigma of mental health and the lack of accessible resources (Deivanayagam et al., 2024).

This issue is mainly common in the Asian region and in Sri Lanka, where mental health support systems are under development. Traditional methods of treatment like in-person meeting appointments with mental health professionals are not accessible to a large portion of the population (Sumathipala, Mahesh & Wijesundara, 2020). Further, the fast expansion of digital platforms has created a growing need for innovative technological solutions to monitor and improve mental health. Integrating Artificial Intelligence (AI) and Machine Learning (ML) in mental health care provides a chance to bridge this gap by providing personalized care, continuous monitoring, and real-time help (Patel et al., 2018).

This project – MoodSync's main aim is to develop a mental health monitoring system with machine learning that uses natural language processing (NLP) and sentiment analysis to track and then predict the mental health state of the users. The system will gather user input like texts, and social media posts and analyze them to detect emotions, moods, and mental health statuses. Through advanced data analytics, personalized recommendations, coping strategies, and real-time alerts for loved ones, this software

aims to enhance mental health support for Sri Lanka in practical, accessible, and anonymous ways.

1.2. Problem Definition

Mental health problems are always complex, subtle, and difficult to detect without professional experience. While self-checking tools do exist, many users do not actively get help, and they face challenges when describing their illness. Additionally, in Sri Lanka, access to professional help is minimal and there is a lack of completed tools that provide unique, data-driven mental health care on a larger scale (Deivanayagam et al., 2024).

The current existing systems for mental health monitoring are very generic, lack personalization, and are not integrated with sentiment analysis or mood predictions. Also, there is a lack of real-time systems that send alerts to loved ones or professionals when users may be in danger. Further, existing systems have failed to provide consistent, personalized coping strategies or motivational content that guide users to manage mental health (Tazarv et al., 2021).

Project MoodSync addresses these challenges by developing a solution that,

1. Uses sentiment analysis and mood tracking to understand users' mental health conditions.
2. Gives personalized coping strategies and health advice based on user input.
3. Provides real-time alerts to loved ones in case of life-threatening situations.
4. It has a chat system for users who are uncomfortable with face-to-face meetings.
5. Includes motivational quotes and data visualization, which helps users to manage their mental health more effectively.

1.3. Project Objectives

The main objectives of this project are,

1. **Sentiment Analysis System** – Implement an advanced machine learning model that will analyze user text inputs to predict sentiment, mood, and to provide details about the user's mental health.

2. **Mood Tracking and Predications** – Implement a system that tracks users' moods over time and provides mood predictions to help detect early signs of mental health diseases.

3. **Personalized Mental Health Recommendations** – Based on the user's mood analysis, the application will display personalized mental health advice, coping strategies, and supporting resources.

4. **Real-Time Alerts for Loved Ones** – Implement a feature that sends real-time alerts to the given emergency contact by the user, when the system detects mood changes that show a crisis.

5. **Anonymous Chat** – This lets users use the system in an anonymous manner, which addresses the main concerns of cultural stigma and fear.

6. **Motivational Content** – Provides the user with motivational quotes, tips, and tricks to help improve mental state and encourage positive thinking.

7. **Data Visualization** – Implement graphs and charts to visualize the represented user data further by providing details for mood patterns, trends, and total mental health progress of users.

8. **Adaption to Sri Lankan Users** – Craft the system to match the needs of the Sri Lankan population, considering local cultural aspects and the mental health challenges unique to own regions to train the model.

By achieving the above-mentioned objectives, Project MoodSync mainly aims to make an accessible, innovative, and data-driven mental health monitoring system that could save lives and promote good mental well-being for people in Sri Lanka and the world.

2. System Analysis

2.1. Facts Gathering Techniques

To make sure that the application is being developed on user needs and to address the problem effectively, it is mandatory to gather existing facts and data. The methods discussed below were used to gather information,

1. **Literature Review** – Thorough research was done on existing mental health applications before developing the software, and currently continuing the research that uses machine learning, sentiment analysis, and mood prediction. Seventeen academic research papers, articles, and case studies were done to understand the project's strengths and weaknesses of the current solution in this domain.

2. **Interviews** – Surveys and interviews were done with mental health professionals and diagnosed users. This helped to identify the user's expectations, needs, and challenges they face in seeking mental health care.

3. **User Feedback** – Feedback was taken by the test users of the system when the application was under development mainly to find the features that would be beneficial and to ensure the system's ease of usability.

4. **Competitive Analysis** – Analyzed the existing mental health apps, mainly the ones that are available in Sri Lanka, which provided the details of features and functionalities that are popular and also lacking in the domain.

By using these above-mentioned methods, I got a clear understanding of the problem, which allowed the system design to match both technical requirements and users' needs.

Doctor Details

- Dr. Saman Weerawardhane – (MBBS) Psychiatrist / Mental Health Specialist.
  National Institute of Mental Health (NIMH)
  NIMH, Mulleriyawa New Town, Angoda

Details of the Local Hospital (Data Collection):

- Mana Suwa Piyasa (මනසුවපියස) – Mental Health Clinic
  Colombo South Teaching Hospital – Kalubowila
  B229 Hospital Rd, Dehiwala-Mount Lavinia 10350

- Survey - CLICK HERE!

- Responses - CLICK HERE!

2.2. Existing System

Currently, as per research, there are different types of mental health monitoring systems in the form of both native to web and mobile platforms. These systems only focus on self-reporting and tracking symptoms. However, many of these platforms are limited to their generic scope and functionalities, lacking real-time analysis, personalized advice, and in providing resources for mental health uplifting (Woodward et al., n.d.).

- **Existing Applications** - Common applications like Calm, Headspace, and Moodfit focus on generic mood tracking, mindfulness, and meditation but do not use modern advanced machine learning techniques to provide unique mental health predictions and responses. (Smith et al., 2023), (Woodward et al., n.d.).

- **Shortcomings** - These systems that were researched and explored had no integration with the other services (ex: Nearby hospitals, Professionals booking, etc.) and it also does not have an alert system for the users experiencing suicidal levels in mental health (Smith et al., 2023), (Tazarv et al., 2021).

- **Cultural / Regional Gap** - Many of the current mental health applications fail to identify the cultural differences and needs they have (ex: post war trauma, Economical conflicts etc.), especially in a country like Sri Lanka which is different from other countries where stigma and language barriers prevent the usage of the systems (Deivanayagam et al., 2024).

While there are existing systems that contribute to basic mental health awareness, they do not address all the needs for personalized, real-time monitoring mainly in the Sri Lankan context.

2.3. Use Case Diagram

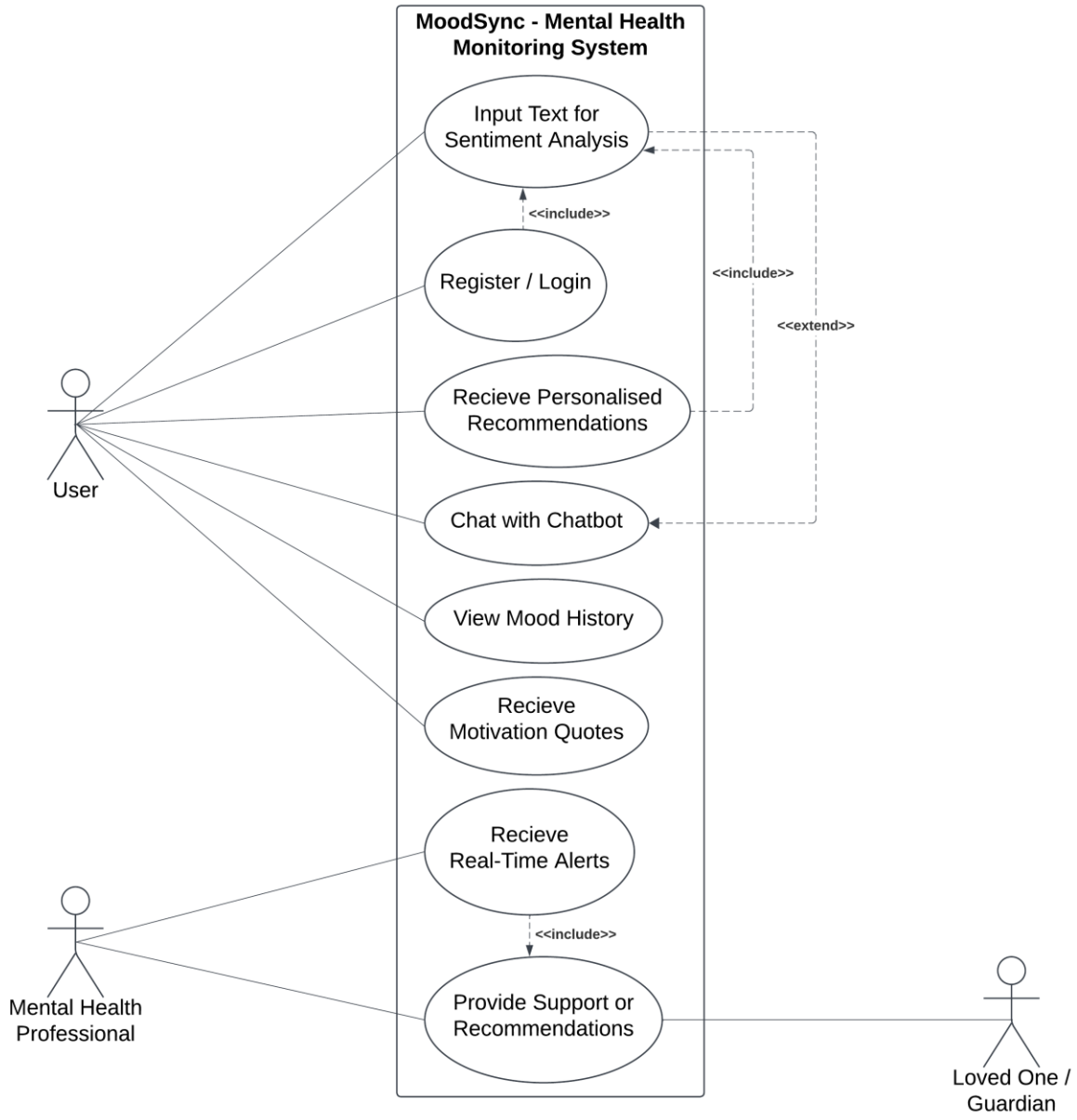


Figure 1 - Use Case Diagram

2.4. Drawbacks of the Existing System

Here are the drawbacks identified in the existing systems,

1. **Lack of Personalization** – The existing systems offer generic advice and do not consider unique mental health patterns or cultural backgrounds. They do not adjust recommendations based on users' unique mental health state.

2. **Limited Interaction** – Many of the existing systems provide only one-way testing methods to get results (ex: health quizzes), where there is limited interaction with the user. Users may benefit from a system if it is more engaging, has real-time support, especially in health breakdowns.

3. **Lack of Real-Time Alerts** – Existing systems lack real-time notification alert features for loved ones/guardians or assigned professionals when the user is in distress. In dangerous situations, immediate support given can be lifesaving, which the system lacks.

4. **No integration with near health care providers** – No current system has integrated mental health professionals or hospitals to assist. This reduces the help in need, which fails the whole product.

5. **Cultural Relevance** – Many of the systems are designed for the global audience where they do not shape to detect cultural differences in mental health understanding (ex: post-war trauma, economic situations, etc.).

This project – MoodSync addresses these issues mentioned with a more prenasalized, real-time, and culturally trained solution with additional features such as anonymous

communication, mental health professional's connections, and real-time alerts for loved ones.

3. Requirements Specification

3.1. Functional Requirements

The functional requirements detail the essential features and functions that this system should have to meet the project objectives as I mentioned in Chapter 1. These requirements ensure that the system delivers the needed features for mental health monitoring and tracking. The functional requirements are,

1. **User Registration and Authentication**

    - The user should be able to create an account and log in securely.

    - User data needs to be stored safely, and privacy needs to be ensured by using encryption.

2. **Sentiment Analysis and Mood Prediction**

    - The system should analyze user inputs (text, social media posts, etc.) to detect user sentiment and predict moods.

3. **Mood Tracking**

- The system should be able to track users' moods over time, which should record any daily input and provide insights into mood patterns.

- Users should be able to view their mood history by graphs and visualization features.

4. **Unique Mental Health Recommendations**

- From sentiment analysis and mood tracking, the system should give unique mental health advice, coping strategies, and resources to improve mental health.

- Recommendations given should be matched to users' emotional and mental health condition.

5. **Anonymous Chat Feature**

- Users should be able to communicate with the system through an anonymous chat interface.

- The chatbot should be able to provide responses, ask questions back, and give guidance based on user inputs.

6. **Real-Time Alerts for Loved Ones**

- The system should automatically send real-time alerts for users' loved ones when the user's mood is tracked as dangerous or self-harming.

- Alerts should be sent by text SMS.

7. Motivational Content

- The application should display motivational content like quotes, affirmations, and positive content to cheer users in difficult times.

8. Data Visualization

- The system must show users' mood data, and sentiment trends by graphs and charts, that allow the user to track their progress over time.

9. Mental Health Professionals and Hospital Connections

- This system should have an option to connect with mental health professionals or the local hospitals.

- This feature should provide contact details and emergency help for mental health support.

3.2. Non-Functional Requirements

Non-functional requirements detail this software's operational characteristics, which ensure that the system performs efficiently and addresses the user's expectations in performance, security, and usability. The non-functional requirements identified are,

1. **Performance**

    - This application should display the real-time analysis when a user inputs the texts and generates recommendations within 5 – 10 seconds.

    - This application should also have the capability to handle many users without running into any issues.

2. **Scalability**

    - This application should be made to scale efficiently as the users increase, which allows further expansion without any performance reduction.

3. **Security**

    - This application should ensure privacy, integrity, and the availability of the user data.

    - Encryption methods should be applied for sensitive data like user login details and personal health information.

4. **Usability**

- This system should have a user-friendly interface to easily navigate mainly for users having less technical knowledge.

- The chat feature should be simple and easy to use.

5. **Availability**

- This system should be 24 * 7, ensuring that users can access mental health monitoring at any time.

- Downtime should be reduced, and it should only happen when the system is maintained only.

6. **Reliability**

- The system should be reliable, giving accurate sentiment analysis, mood tracking, and recommendations on user inputs.

- The system should handle unexpected errors so that the user experience will not be disturbed.

7. **Maintainability**

- This system should be easy for maintenance purposes, with components.

8. **Compliance**

- This system must follow Sri Lankan data protection regulations (2022) like GDPR and privacy laws to ensure that user data are handled ethically.

3.3. Hardware / Software Requirements

The hardware and software requirement mentioned shows the resources that are needed to develop, deploy, and run the software. This includes server needs, development tools, and platforms for the deployment.

1. Hardware Requirements

- **Server** – A cloud server (ex: AWS) to host the application backend services, machine learning models, and the database.

- **End User Devices** – The system should be responsive to work on web applications from desktops and smartphones.

- **Database Storage** – Cloud-based database storage needed (ex: MongoDB) for storing the user data securely.

2. Software Requirements

- **Frontend Development** – React.js, Vite, TypeScript, Tailwind CSS, and Flowbite library for styling.

- **Backend Development** – FastAPI with Python, Node.js with Express for the API endpoints and to handle machine learning processes.

- **Machine Learning Libraries** – Pandas, scikit-learn, imblearn, NLTK, and other Python libraries are mainly used for sentiment analysis and to predict moods.

- **Database** – MongoDB for storage of user data, feedback, mood logs, and history.

- **Cloud Hosting / Platforms** – AWS for handling computational tasks and hosting the application.

3.4. Networking Requirements

This project has less network requirements, but these are the basic requirements for system components and external services communication in the application,

1. **Internet Connectivity**

    - This system needs a good internet connection to provide real-time updates and alerts.

    - Software must be able to communicate with the backend server to process the data.

2. **API Integration**

    - The system needs a third-party API integration for the SMS notifications and integration with healthcare providers.

3. **Real-Time Communication**

    - For the real-time alerts feature, the system should use push notifications by WebSockets (Socket.IO) or a similar way to ensure delivery of the alerts.

4. Feasibility Study

Carrying out a feasibility study is needed for practicality and survival of the product. This chapter details the operational, technical and financial aspects of the project MoodSync, ensuring that this project can be further developed and completed with the limits of given time, resources and the budget.

4.1. Operational Feasibility

Operational feasibility mainly shows whether this proposed system can be implemented effectively and used in the real world, considering the user's needs, accessibility, and system integration.

1. User Adoption

- This system targets users going through mental health challenges, especially in a low-accessed country like Sri Lanka. It mainly focuses on good free software for people who are hesitant/shy/feared to look out for help in need. Anonymous chat and real-time alert features for loved ones increase the user's adoption of this application.

- This system will be designed with a good, featured user interface, matching for the users who have less technical knowledge. This improves the ease to use this software, and it also make sures that users can access this system well.

2. **Integration of mental health professionals**

- One of the key features of this system is the connection with mental health professionals. This feature makes sure that users have the professional's quick access when needed.

- By giving easy access to professionals and hospitals, this system assures that users have a solid network to support, improving this system's operational possibility.

3. **Cultural Adoption**

- This system will be developed mainly to meet the needs of Sri Lankan users, where features will be included with well-tuned cultural context and sensitivities.

- This system's design and functionality consider cultural details related to mental health, reducing its stigma and encouraging people to get help.

4. **Real-Time Alerts**

- Real-time alerts for loved ones will provide operational advantages, ensuring quick actions in case of a mental health crisis. This driven feature helps users during critical moments by reducing life-threatening situations.

- These alerts especially benefit the user who will not have quick access to healthcare providers, ensuring their safety and well-being on top.

4.2. Technical Feasibility

Technical feasibility shows if the technology required is available, reliable, and matching to successfully develop the project's solution.

1. Technology Stack

- This system uses modern, scalable technologies which are well suited to developing a system with mental health monitoring. Using MERN Stack with TypeScript for the frontend (client) and FastAPI and Node.js for the backend (server) gives an efficient solution for developing the web application for the system.

- The Python-based machine learning model uses popular libraries like scikit-learn, Pandas, and NLTK mainly for mood predictions and sentiment analysis. This tech stack is reliable and proven to build good models under different machine learning and data analysis software.

2. Machine Learning and NLP

- This system's need to analyze the user inputs and predict mood by sentiment analysis requires advanced Natural Language Processing (NLP) techniques. Python libraries such as TF-IDF and logistic regression have good documentation to refer, and it is efficient for text classification tasks.

- The SMOTE algorithm will be used to handle the dataset’s imbalances, making sure that the model performs well even when some classes (Ex: mental health disorders) are not present in the trained dataset.

3. **Cloud Infrastructure**

- This system will be hosted on the AWS (Amazon Web Services) cloud platform EC2 Service (free tier) to temporarily provide the application to users.

- AWS EC2 hosting provides higher availability, a secure database, and flexibility in accessing the software.

4. **Security and Data Protection**

- Given the sensitive domain of health data, modern strict security measures will be used like data encryption in chats, chat protocols and safety practices to protect user privacy. (ex: OAuth 2.0, JWT, Bcrypt).

- This system will match the industry’s best practices to ensure that the user's data is kept secure.

4.3. Outline Budget

This project – MoodSync does not include any additional charges and given charges would be taken care of by my personal funds.

1. **Mental Health Doctor** – Appointment charge of Rs. 5,500 (£14.85) for data gathering and an additional Rs. 1000 (£2.62) for every visit to check the application quality changes and inspections.

2. **Development** – Gets paid versions of development tools free facilitated by University of Plymouth student account for JetBrains, PyCharm Ultimate Edition IDE.

3. **Cloud Hosting** – Use of free tier of cloud services (AWS) at present, domain and SSL certificate and charges will be applied for further hosting.

5. System Architecture

- Link for all diagrams for a clear view - CLICK HERE!

5.1. Class Diagram of Proposed System

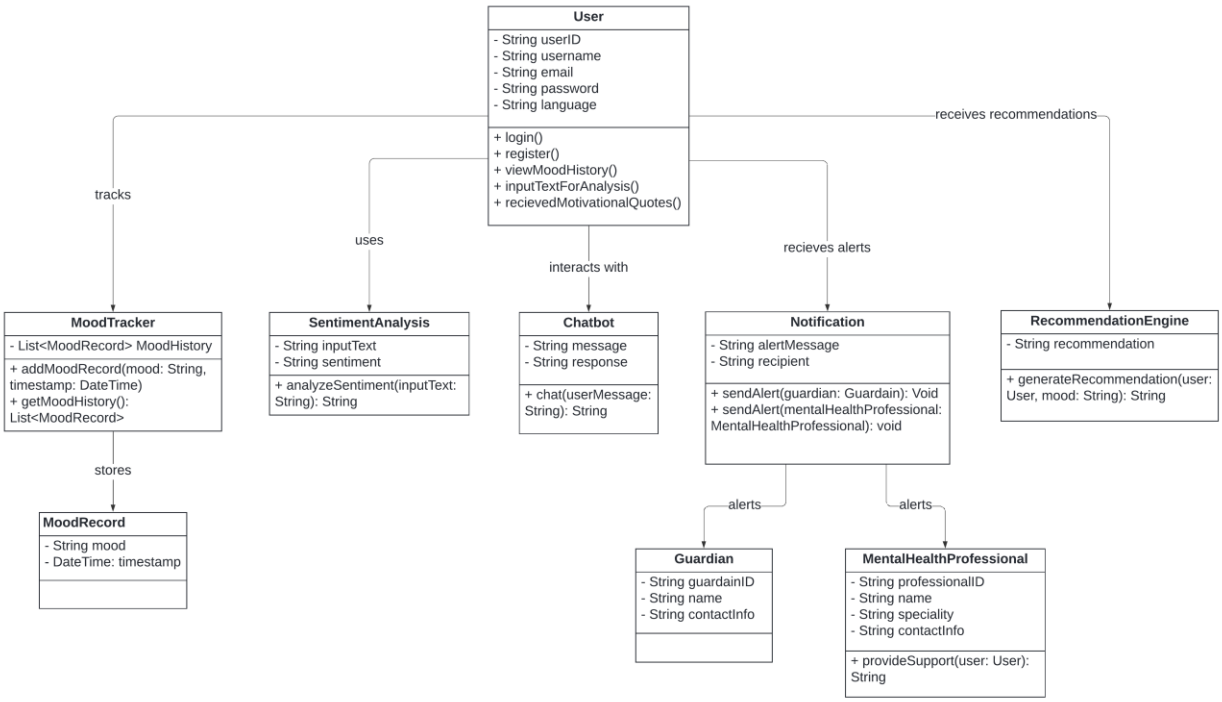


Figure 2 - Class Diagram

5.2. Database Diagram

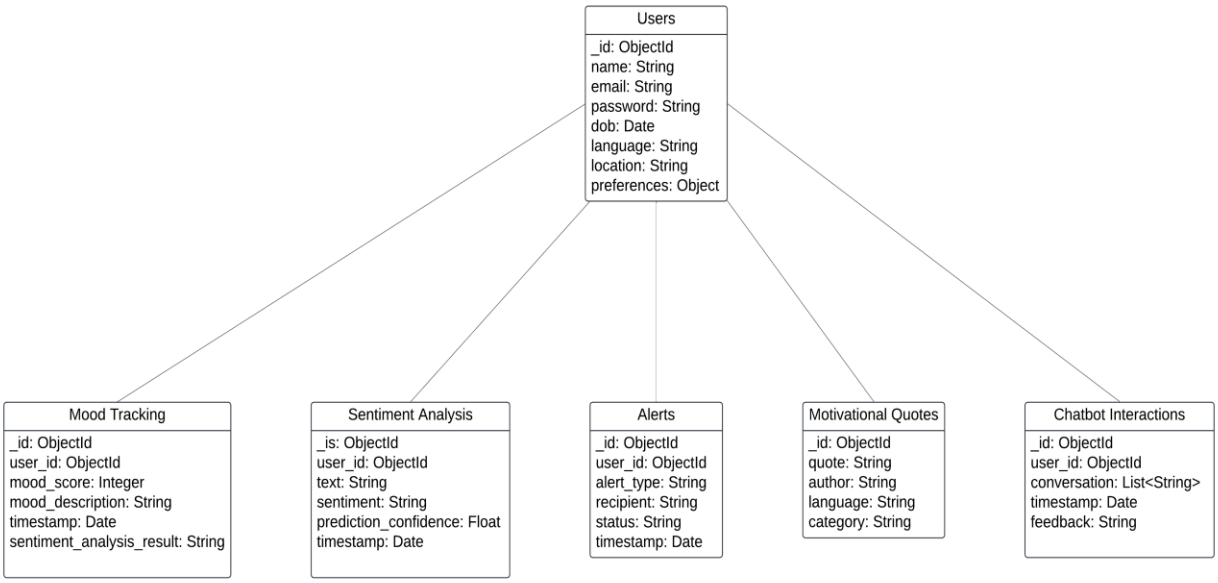


Figure 3 - Database Design

- ***NOTE*** *– ER Diagram cannot be drawn for this software project as it uses NoSQL unstructured database. The database design diagram is drawn above showing how the data collections are managed, and how their connections interact.*

5.3. High – Level Architectural Diagram

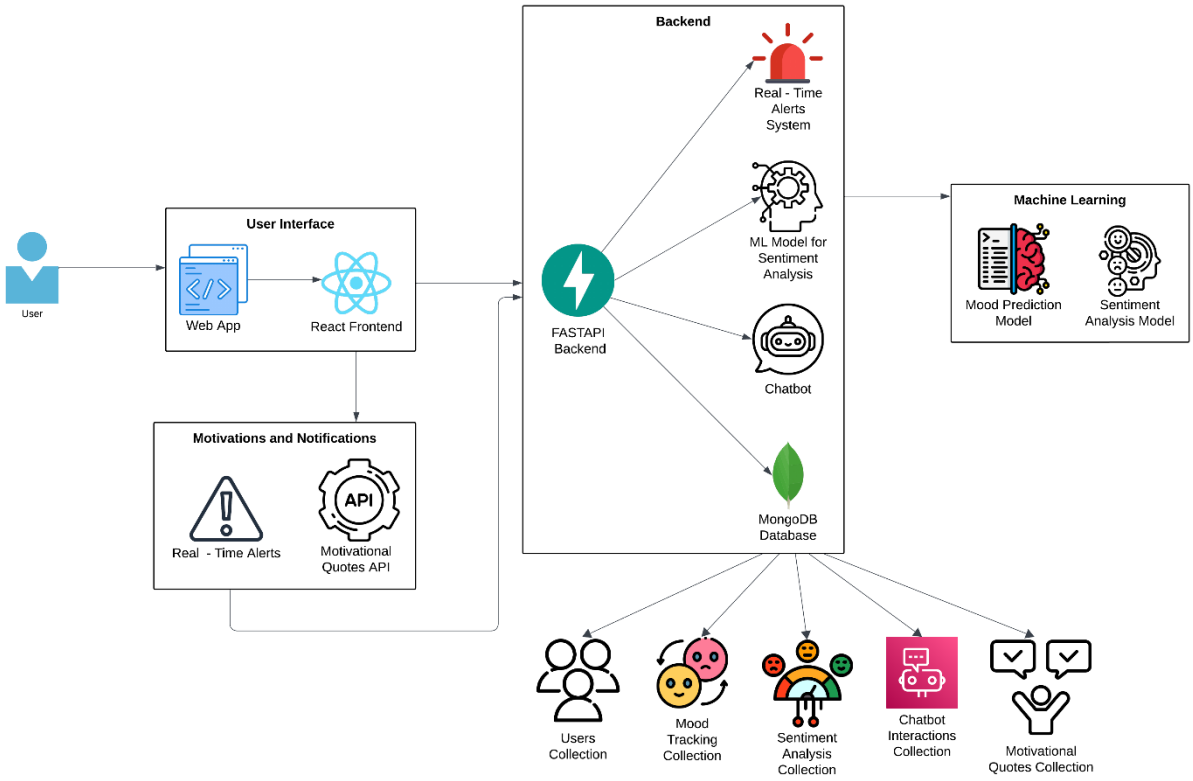


Figure 4 - High - Level Architectural Diagram

5.4. Networking Diagram

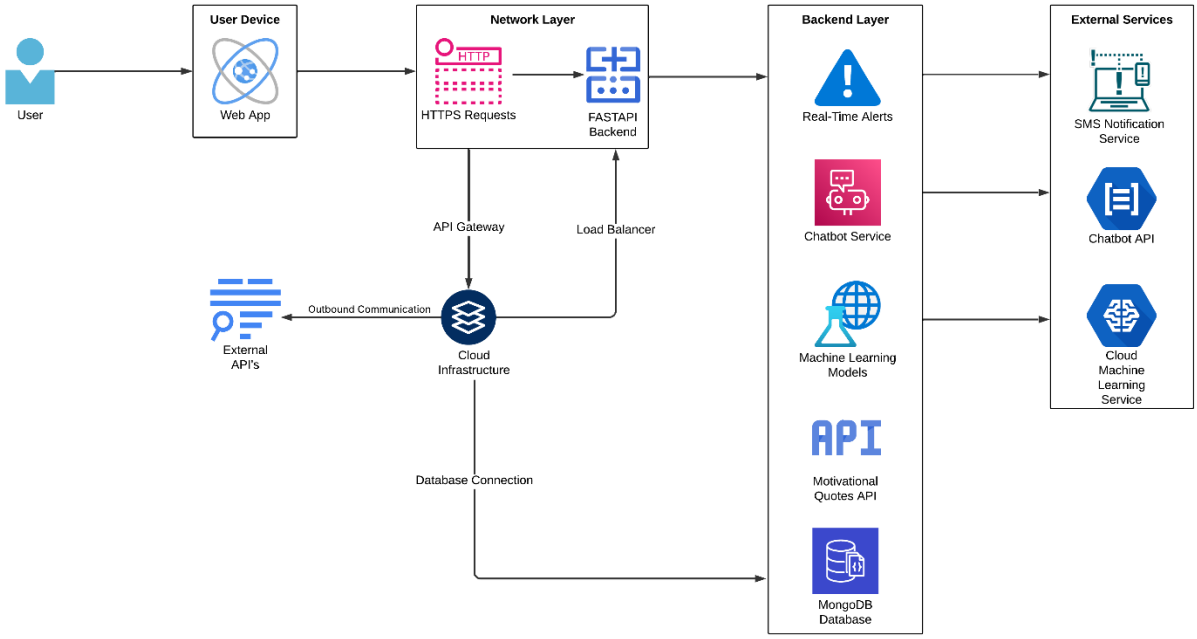


Figure 5 - Networking Diagram

6. Development Tools and Techniques

The development of the project MoodSync has been done with modern technologies and tools to create a robust, adaptable and user-friendly application. This chapter shows the main tools, libraries, frameworks and methodologies used in the development of this software. These tools and technologies were chosen to ensure that the application matches the given functional and non-functional requirements while providing good user experience.

6.1. Development Methodology

The development methodology selected for this project is Agile, though this is an individual project, it allows a flexible and repetitive process. The agile methodology focuses on delivering functional progress of the application regularly, with continuous feedback from the test users and mental health professionals which is considered for the development process. By using this approach, it helps to ensure that this software is developed based on the real user's needs and can suit the change of requirements over time.

- **Sprint-Based Development** – This project is divided into sprints, which last for 2 weeks. Every sprint is focused on separate functionalities, like user authentication, sentiment analysis or mood-based tracking.

- **Continuous Integration and Testing** – Agile prioritize the importance of testing in the development process itself. Automated testing tools and continuous integration pipelines ensure that the new features are tested frequently and integrated into the application.

- **User Feedback** – After every sprint, feedback is taken from a set of test users to make sure that the application follows the expectations and solves the expected problem effectively.

6.2. Programming Languages and Tools

This system is built using different programming languages and tools to ensure that both frontend and backend components work smoothly, where machine learning is used for sentiment analysis and mood prediction.

Frontend Development

- **React.js** – React.js is a JavaScript library that is used to build the user interfaces (UI) for this project MoodSync. Reacts allow the development of dynamic, responsive & interactive UIs where components can be reused (ex: Navigation Bar, Footer etc.). It is a fast and efficient way to develop maintainable and scalable software.

- **Vite** – Vite is used as for both the development server and the build tool. This ensures fast hot module reload (HMR) and builds optimization, which improves the development flow, making it fifteen times faster than using react alone.

- **TypeScript** – TypeScript is used for modern frontend and backend development to add static typing, which will reduce errors and improve the quality by ease of catching bugs in the development stage itself.

- **Tailwind CSS** – TailiwndCSS is mainly used for front-end styling. The first utility CSS framework that allows for a fast, responsive, and user-customized design. It simplifies the design processes by giving a set of classes that can be used with HTML.

- **Flowbite** – Flowbite is a component UI library used with the Tailwind CSS framework that gives pre-designed, responsive UI components like modals, buttons, and form controls, which makes the UI development faster and more consistent.

Backend Development

- **FastAPI** – Is used to develop the main backend of the software. It is the latest fast-performance web framework for building APIs with the use of Python. It supports auto-generation of the API documents and also it comes up with built-in features like dependency injection, making it more of a best choice for developing RESTful APIs. Also, its support for asynchronous programming makes sure that the backend will handle multiple requests at the same time, improving performance and scalability.

- **Python** – Python is the main programming language used in this project which is for backend and machine learning developments in the software. Python is mainly used in data analytics, machine learning, and web development, which makes it the best choice for this project.

- **Node.js** – Node.js backend is also used apart from FastAPI for handling normal third-party APIs (ex: Motivational Content) as some APIs don't support python environments. Express.js will also be used as the middleware.

Database

- **MongoDB** – A NoSQL database used for this project to store users' data, mood tracking records, and the software logs. It is mainly chosen for its scalability, flexibility, and easy integration with the software. It allows the software to store unstructured data which will be essential to handling different dynamic natures of user input data.

Machine Learning

- **scikit-learn** – The scikit-learn library is used for machine learning development in this software. It provides efficient tools for sentiment analysis and mood prediction model building. The logistic regression model in scikit-learn is used to separately identify sentiments from users' text inputs and help in predicting mental health conditions.

- **Pandas** – Used for data manipulation and preprocessing. It does provide data structures that allow to handle data efficiently and for transformation, which is important for purifying and using the input data for the machine learning models.

- **NLTK** – The Natural Language Toolkit (NLTK) is used for text preprocessing, including tokenization, lemmatization and to remove the stopwords. These NLP methods help in normalizing and cleaning the user input data, which does a big role in improving the accuracy of sentiment analysis and mood predictions.

- **Imbalanced-learn (imblearn)** – This learning library is used mainly for handling dataset imbalances with the usage of techniques like SMOTE (Synthetic Minority Oversampling Technique) which ensures that the machine learning model performs well when the mental health conditions are not fully represented in the dataset that is used to train the model.

6.3. Third Party Components and Libraries

To improve the functionalities and efficiency of this application, third-party libraries and tools have been used to build this application. These below-mentioned components allow the software to have advanced features and largely reduce its development time.

1. **Twilio (SMS Notification)** – This is integrated with the application to send real-time SMS alerts to users' loved ones and mental health professionals. This software uses this API to send SMS messages when triggered automatically by the system, making sure that alerts are delivered quickly.

2. **Chart.js** – This is used to create interactive and dynamic data visualization. It helps to display users' tracking details and sentiment analysis results display in graphs, where users and doctors can use them to understand their mental health over time.

3. **Socket.IO** – This is used for real-time communication between the backend (server) and the frontend (client) which lets features like messaging in the chatbot and real-time alerts to users' loved ones and to communicate to health professionals.

4. **Passport.js** – This is used for user authentication. It supports many authentication methods such as local login, email-based login, and OAuth2.0 making it an easier task to develop secure and custom authentication methods.

5. **Bcrypt** – This is used mainly for password hashing purposes. It ensures that the user's password is stored safely, and it also cuts off unauthorized access to sensitive information.

6.4. Algorithms

1. **Logistic Regression** – This is used as the primary algorithm to classify sentiments from user input. It helps to predict the mental state of the user by text data that is collected from text input or by chat.

2. **TF – IDF (Term Frequency – Inverse Document Frequency)** – It is mainly used to convert text data to numerical features so that it can be easily processed in the trained machine learning models. It also measures the importance of words in context by the user's input and helped to improve the performance of sentiment analysis.

3. **SMOTE (Synthetic Minority Oversampling Technique)** – This is applied to the dataset to handle class imbalances. It generates synthetic examples in the minority class, making sure that the model can learn well from all given classes.

4. **K–Fold Cross Validation** – This is used to evaluate the machine learning model performance. The dataset will be split into K subsets, where training and testing will happen K times to ensure accuracy and adaptability.

7. Discussion

7.1. Overview of the Interim Report

This interim report shows the current progress and state of the Project MoodSync. This project aims to provide innovative and accessible software to monitor users' mental health through sentiment analysis, mood tracking, and personalized recommendations. By using machine learning technologies, the system tracks mental health patterns by user inputs and gives real-time intervention when needed. This report also covers the system’s requirements, feasibility analysis and technologies that were used to build this software.

7.2. Summary of the Report

This report provides an in-depth analysis of the problem in mental health monitoring, mainly in the Sri Lankan context. It shows the system’s functional and non-functional requirements, also the technological needs to support this solution. It details different challenges the applications face, like lack of personalization, real-time intervention, and cultural inclusivity, which this project addresses. Additionally, the report details the tools, technologies, and machine learning algorithms used in this application, ensuring an software that is a secure and adaptable solution for the users.

7.3. Challenges Faced

In the development of this software, some challenges were faced like,

1. **Data Quality and Preprocessing** – The input dataset used to train the model needed thorough cleaning and preprocessing to ensure that it was suitable for mood predictions and sentiment analysis. Balancing the dataset using methods like SMOTE was essential to improve the model's output accuracy.

2. **Real-Time Alerting** – Developing a reliable real-time alert system for loved ones was a challenge, mainly to ensure that the notifications were delivered in real-time effectively processed within backend without a delay.

3. **Cultural Adaption** – Shaping the system to maintain high accuracy in sentiment analysis was a major technical challenge with the cultural inclusions, which needed to include regionally native text processing methods.

4. **User Privacy** – Making sure that the data stored and processed are without changed privacy concerns was challenging, which needed robust encryption methods (ex: OAuth 2.0, JWT & Bcrypt) and followed with Sri Lankan data protection acts.

7.4. Future Plans / Upcoming Work

The next further steps in the project is to include,

1. **Model Optimization** – To tune the machine learning model further to improve its accuracy and robustness, mainly in managing real-time users' data and different types of inputs.

2. **Feature Enhancement** – Adding extra features like integration with wearable devices (for tracking physical and health data) and expanding the emergency alert system for more precise locations for better user safety.

3. **Deployment** – Prepare the application to publish on a large scale, with a focus on cloud integration to make sure it can handle a growing user base concurrently providing real-time support without running into performance issues.

4. **User Testing** – Keep up through the user testing, to get feedback on the application's core usability and effectiveness, and make sure that it follows the user's expectations.

References

- Deivanayagam, T.A., Ní Chobhthaigh, S., Devakumar, D., Patel, K. & Rannan-Eliya, R.P. (2024) Mental health prevalence, healthcare use and access between 2018 and 2022 in Sri Lanka: an analysis of survey data. *Global Health Research*. 1–16. doi:10.3310/hjwa5078.

- Patel, V., Saxena, S., Lund, C., Thornicroft, G., Baingana, F., et al. (2018) The Lancet Commission on global mental health and sustainable development. *Lancet (London, England)*. 392 (10157), 1553–1598. doi:10.1016/S0140-6736(18)31612-X.

- Smith, K.A., Blease, C., Faurholt-Jepsen, M., Firth, J., Van Daele, T., Moreno, C., Carlbring, P., Ebner-Priemer, U.W., Koutsouleris, N., Riper, H., Mouchabac, S., Torous, J. & Cipriani, A. (2023) Digital mental health: challenges and next steps. *BMJ mental health*.26 (1). doi:10.1136/bmjment-2023-300670.

- Sumathipala, A., Mahesh, D. & Wijesundara, H. (2020) *Challenges of Research in Rural Mental Health in Sri Lanka*. In: pp. 195–220. doi:10.1007/978-981-10-2345-3_30.

- Tazarv, A., Labbaf, S., Reich, S.M., Dutt, N., Rahmani, A.M. & Levorato, M. (2021) Personalized Stress Monitoring using Wearable Sensors in Everyday Settings. In: *2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. 2021 pp. 7332–7335. doi:10.1109/EMBC46164.2021.9630224.

- Woodward, K., Kanjo, E., Brown, D.J., Mcginnity, T.M., Inkster, B., Macintyre, D.J. & Tsanas, A. (n.d.) *Beyond Mobile Apps: a Survey of Technologies for Mental Well-being*.

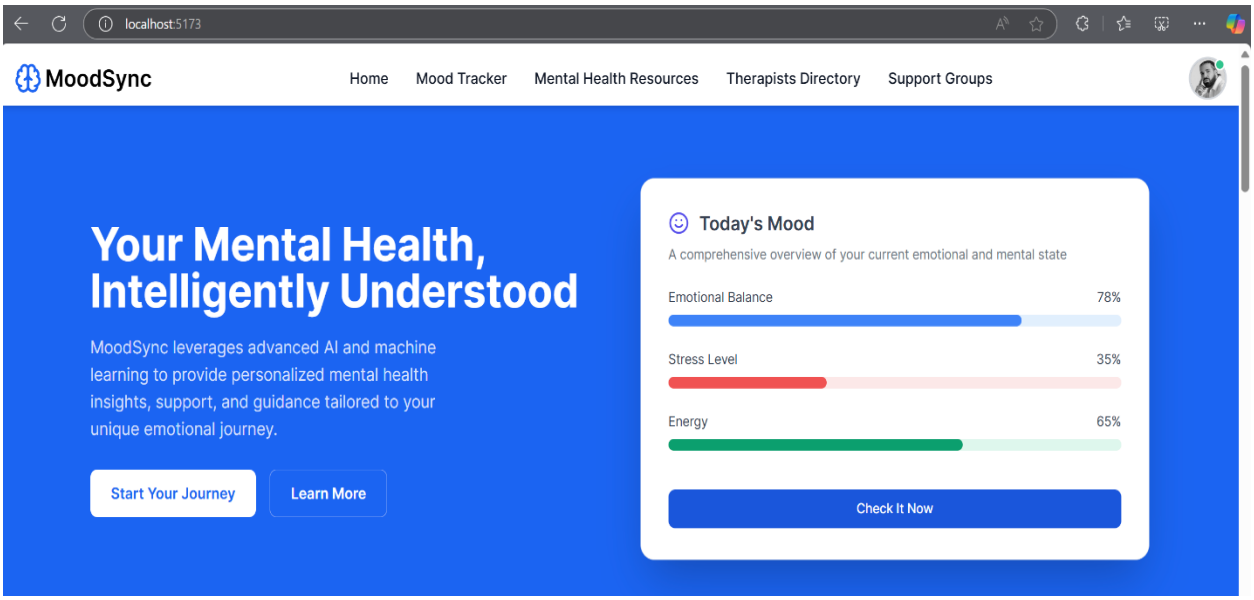
- Calm - https://www.calm.com/
- Headspace - https://www.headspace.com/
- Moodfit - https://www.getmoodfit.com/

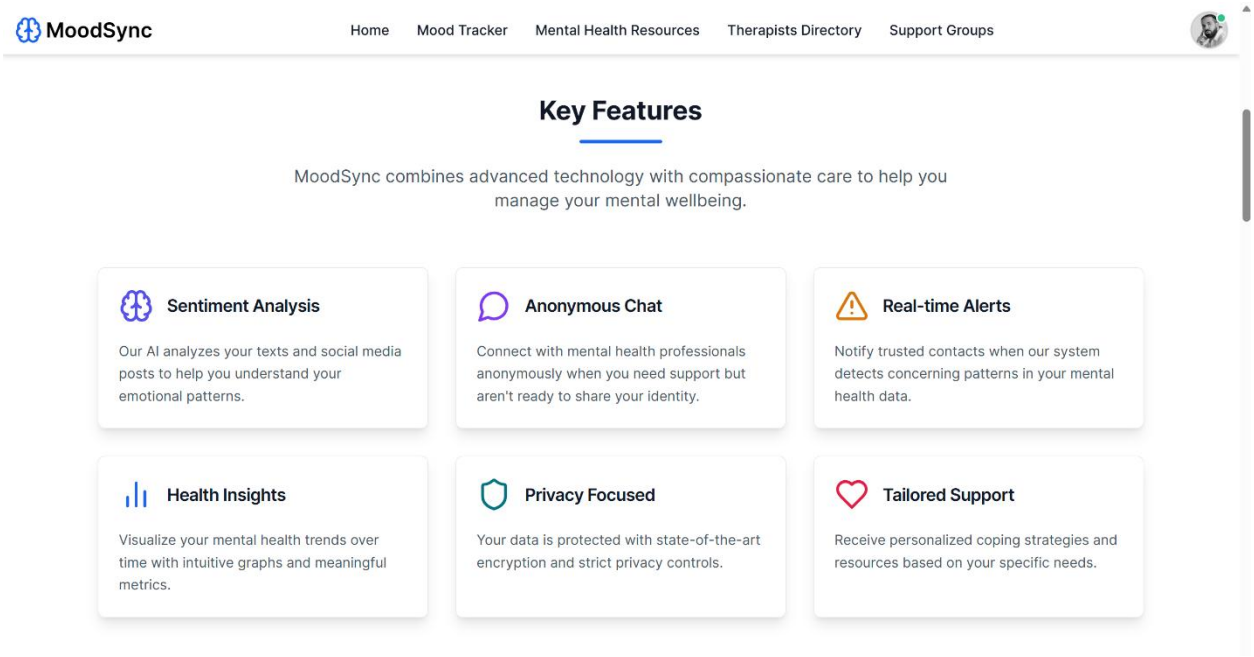
Appendix

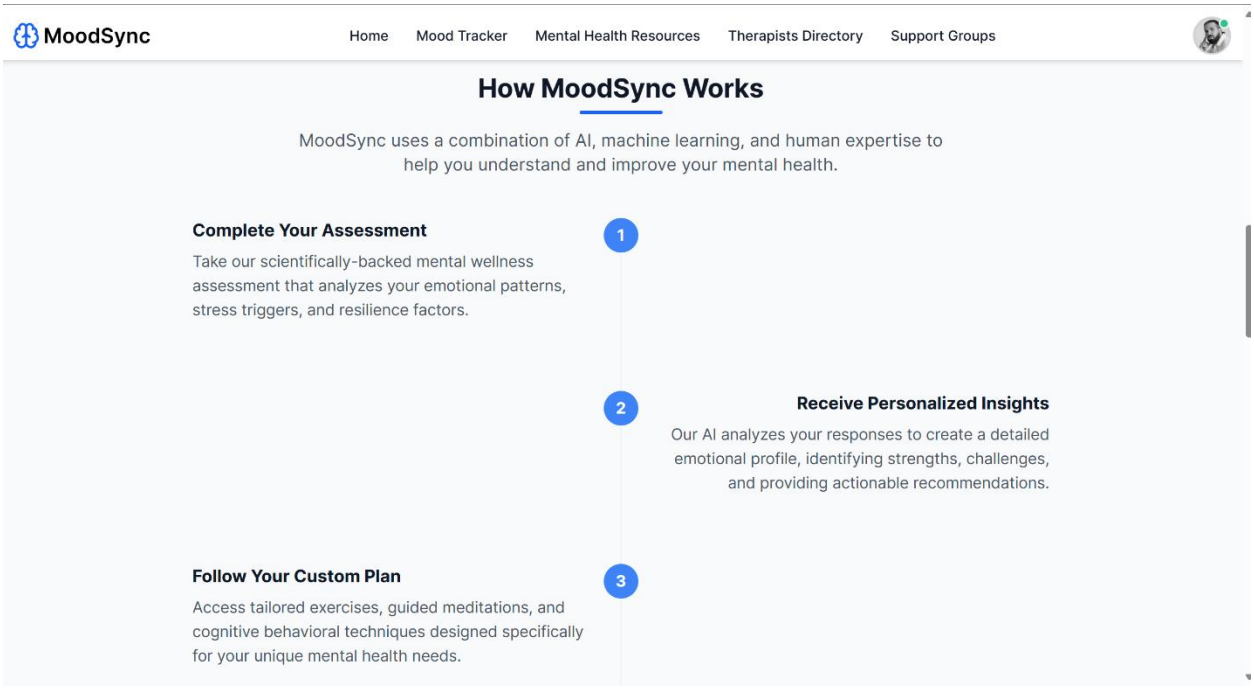
All Links

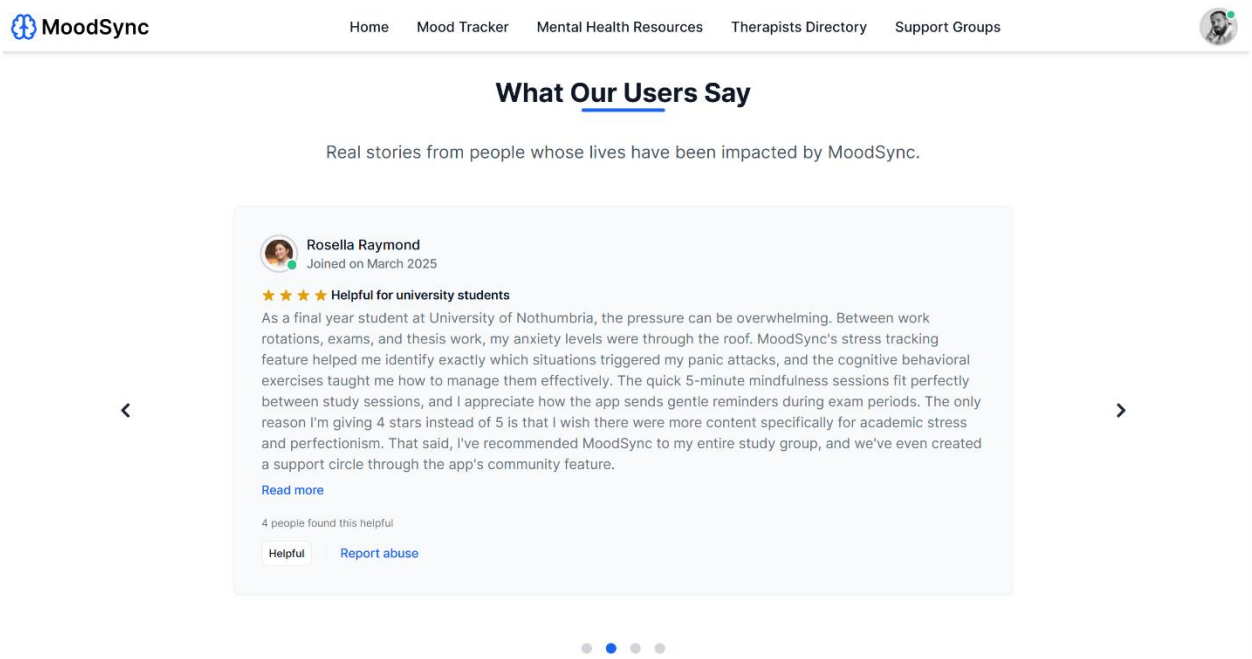
- Diagrams - CLICK HERE!
- Survey - CLICK HERE!
- Responses - CLICK HERE!

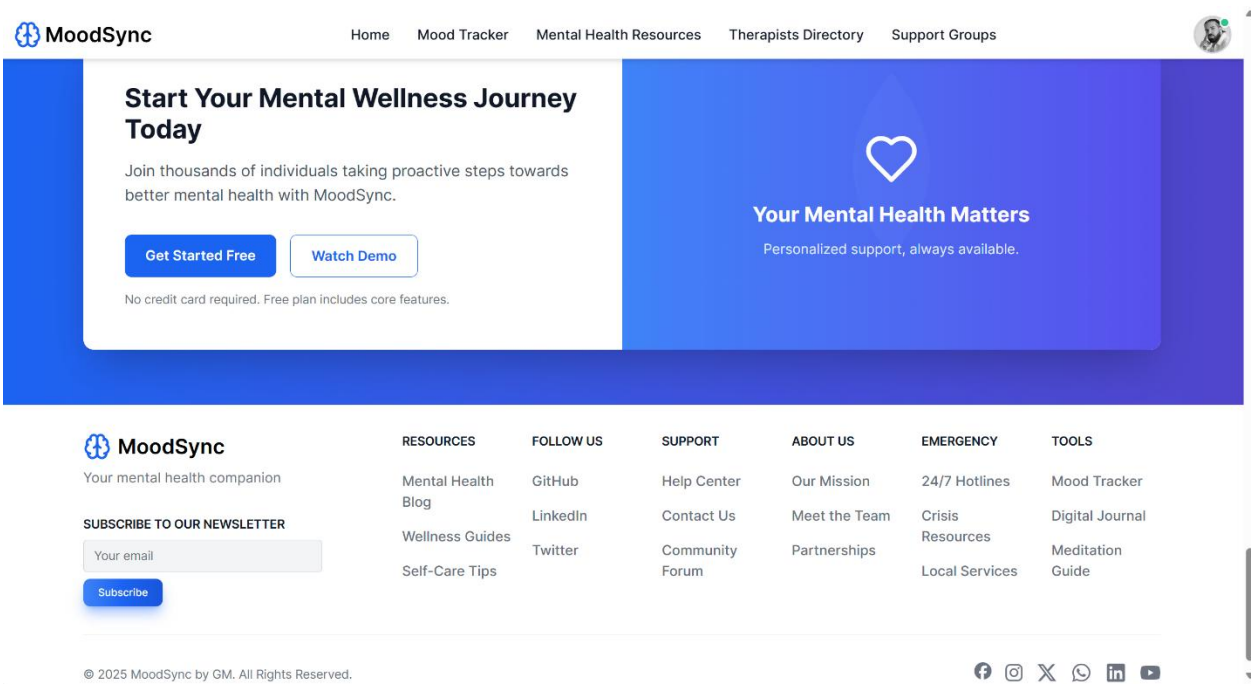
User Interfaces

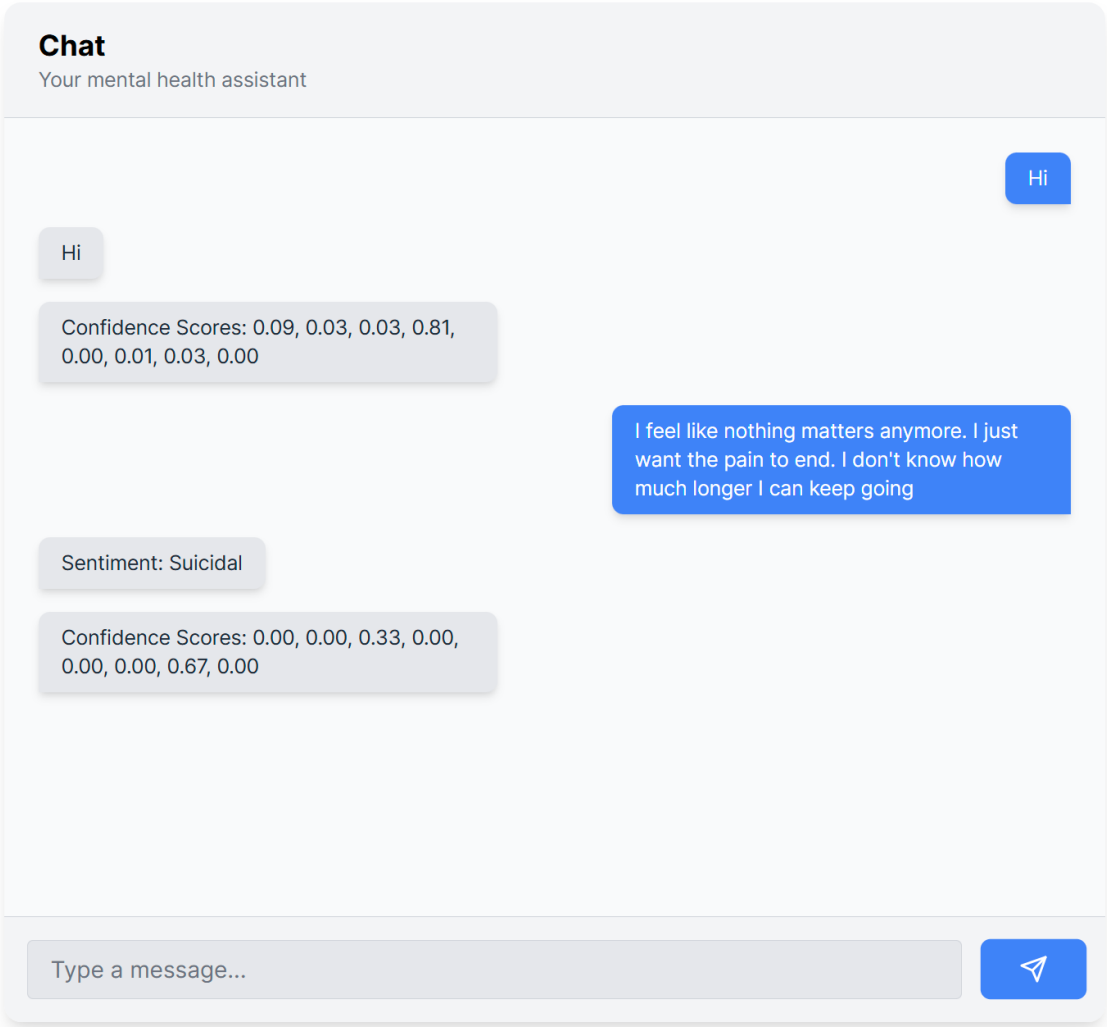



MoodSync
Home
Mood Tracker
Mental Health Resources
Therapists Directory
Support Groups
Key Features
MoodSync combines advanced technology with compassionate care to help you manage your mental wellbeing.
Sentiment Analysis
Our AI analyzes your texts and social media posts to help you understand your emotional patterns.
Anonymous Chat
Connect with mental health professionals anonymously when you need support but aren't ready to share your identity.
Real-time Alerts
Notify trusted contacts when our system detects concerning patterns in your mental health data.
Health Insights
Visualize your mental health trends over time with intuitive graphs and meaningful metrics.
Privacy Focused
Your data is protected with state-of-the-art encryption and strict privacy controls.
Tailored Support
Receive personalized coping strategies and resources based on your specific needs.


MoodSync
Home
Mood Tracker
Mental Health Resources
Therapists Directory
Support Groups
How MoodSync Works
MoodSync uses a combination of AI, machine learning, and human expertise to help you understand and improve your mental health.
1
Complete Your Assessment
Take our scientifically-backed mental wellness assessment that analyzes your emotional patterns, stress triggers, and resilience factors.
2
Receive Personalized Insights
Our AI analyzes your responses to create a detailed emotional profile, identifying strengths, challenges, and providing actionable recommendations.
3
Follow Your Custom Plan
Access tailored exercises, guided meditations, and cognitive behavioral techniques designed specifically for your unique mental health needs.

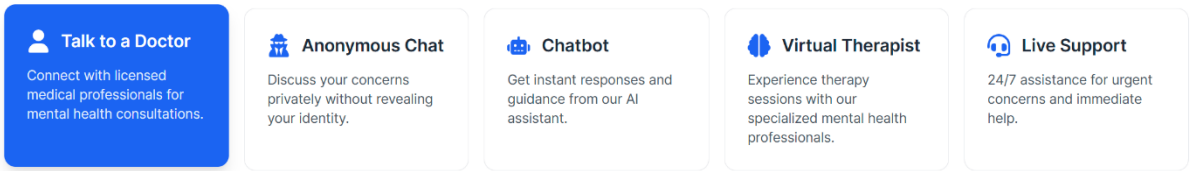

MoodSync
Home
Mood Tracker
Mental Health Resources
Therapists Directory
Support Groups
What Our Users Say
Real stories from people whose lives have been impacted by MoodSync.
Rosella Raymond
Joined on March 2025
Helpful for university students
As a final year student at University of Nothumbria, the pressure can be overwhelming. Between work rotations, exams, and thesis work, my anxiety levels were through the roof. MoodSync's stress tracking feature helped me identify exactly which situations triggered my panic attacks, and the cognitive behavioral exercises taught me how to manage them effectively. The quick 5-minute mindfulness sessions fit perfectly between study sessions, and I appreciate how the app sends gentle reminders during exam periods. The only reason I'm giving 4 stars instead of 5 is that I wish there were more content specifically for academic stress and perfectionism. That said, I've recommended MoodSync to my entire study group, and we've even created a support circle through the app's community feature.
Read more
4 people found this helpful
Helpful
Report abuse

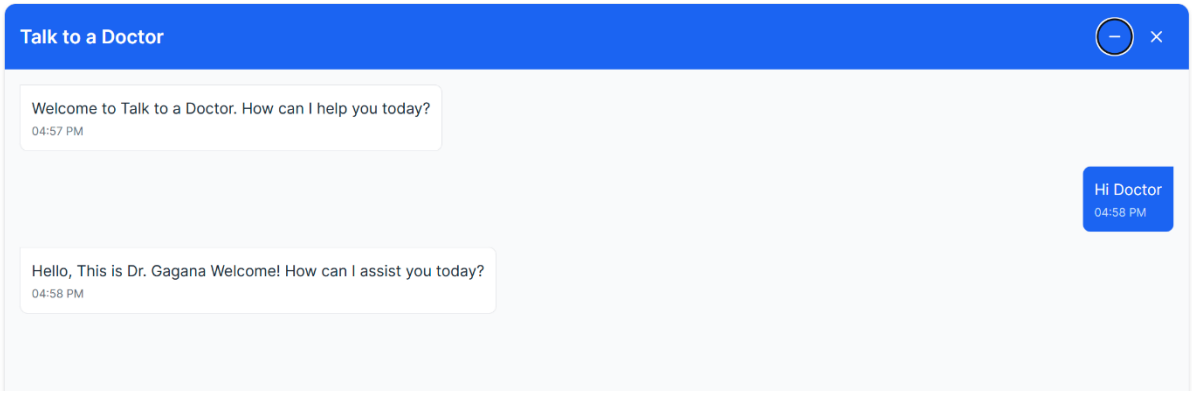

MoodSync
Home
Mood Tracker
Mental Health Resources
Therapists Directory
Support Groups
Start Your Mental Wellness Journey Today
Join thousands of individuals taking proactive steps towards better mental health with MoodSync.
Get Started Free
Watch Demo
No credit card required. Free plan includes core features.
Your Mental Health Matters
Personalized support, always available.
MoodSync
Your mental health companion
SUBSCRIBE TO OUR NEWSLETTER
Your email
Subscribe
RESOURCES
Mental Health Blog
Wellness Guides
Self-Care Tips
FOLLOW US
GitHub
LinkedIn
Twitter
SUPPORT
Help Center
Contact Us
Community Forum
ABOUT US
Our Mission
Meet the Team
Partnerships
EMERGENCY
24/7 Hotlines
Crisis Resources
Local Services
TOOLS
Mood Tracker
Digital Journal
Meditation Guide
© 2025 MoodSync by GM. All Rights Reserved.


Chat
Your mental health assistant
Hi
Hi
Confidence Scores: 0.09, 0.03, 0.03, 0.81, 0.00, 0.01, 0.03, 0.00
I feel like nothing matters anymore. I just want the pain to end. I don't know how much longer I can keep going
Sentiment: Suicidal
Confidence Scores: 0.00, 0.00, 0.33, 0.00, 0.00, 0.00, 0.67, 0.00
Type a message...

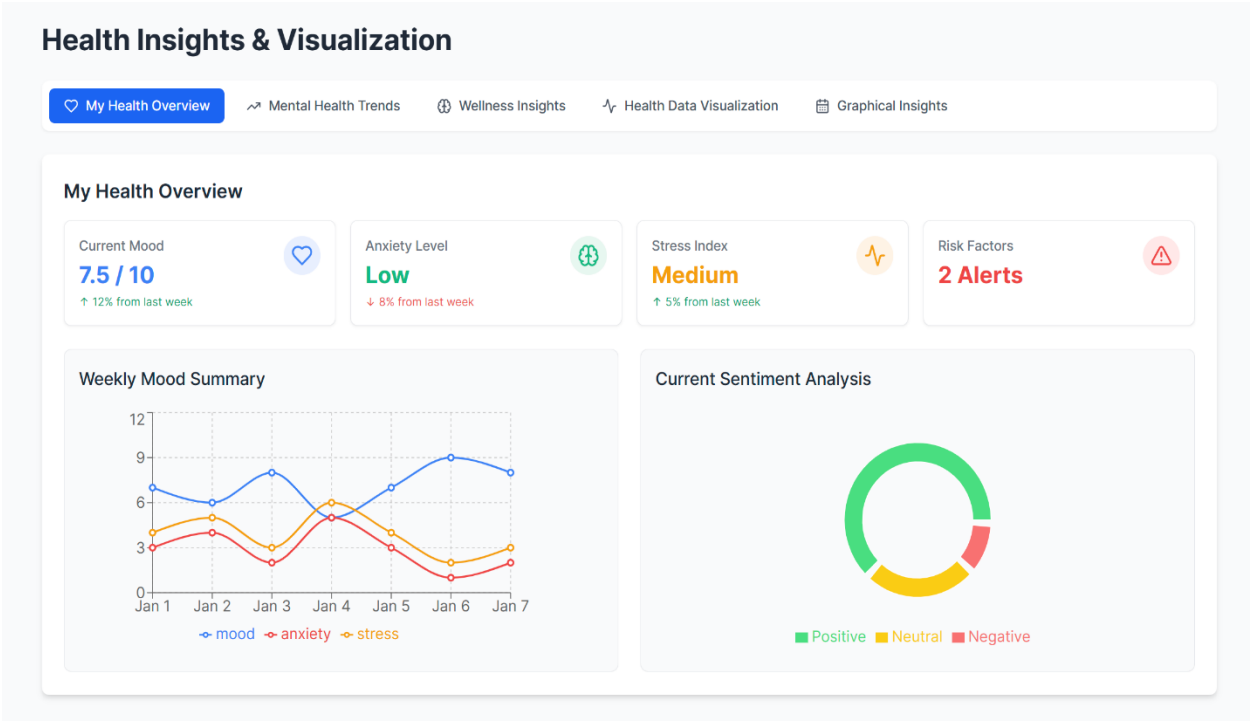
Chat Support Services

Connect with mental health professionals, get support, or use our AI-powered tools for guidance. All conversations are secure and confidential.





Health Insights & Visualization



Patient Management

Search patients...

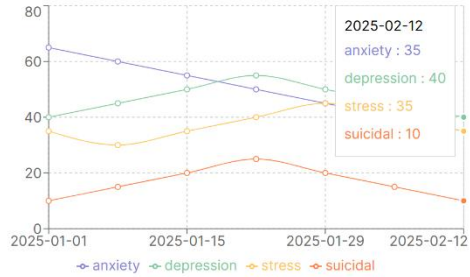
NAME	STATUS	RISK LEVEL	LAST ACTIVE	ACTIONS
Amal Perera	Suicidal	High	2 hours ago	View Message Alert
Malini Silva	Depression	Medium	1 day ago	View Message Alert
Rohan Fernando	Anxiety	Low	5 hours ago	View Message Alert
Dinesh Kumar	Bipolar	Medium	3 days ago	View Message Alert
Priya Mendis	Depression	High	1 hour ago	View Message Alert

MoodSync

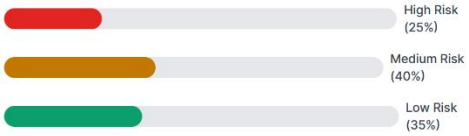
- Dashboard
- Patients
- Inbox
- Trends

Mental Health Trends

Overall Sentiment Distribution



Risk Level Distribution



Machine Learning Model Performance

[8]:
```
# Splitting Dataset.
X_train, X_test, y_train, y_test = train_test_split(X_resampled, y_resampled, test_size=0.2, random_state=42)
```

Hyperparameter Tuning

[9]:
```
# Hyperparameter Tuning for Logistic Regression.
param_grid = {
    'C': [0.1, 1, 10],
    'solver': ['lbfgs', 'liblinear'],
    'penalty': ['l2'],
}
grid_search = GridSearchCV(LogisticRegression(max_iter=1000), param_grid, cv=5, scoring='accuracy')
grid_search.fit(X_train, y_train)

best_model = grid_search.best_estimator_
print("\nBest Parameters:", grid_search.best_params_)
```

```
Best Parameters: {'C': 10, 'penalty': 'l2', 'solver': 'liblinear'}
```

Evaluate Model

[10]:
```
# Evaluate Model.
y_pred = best_model.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)
print("\nAccuracy of the model:", accuracy * 100, "%")
print("\nClassification Report:")
print(classification_report(y_test, y_pred))
print("\nConfusion Matrix:")
print(confusion_matrix(y_test, y_pred))
```

```
Accuracy of the model: 96.78671355400964 %
```

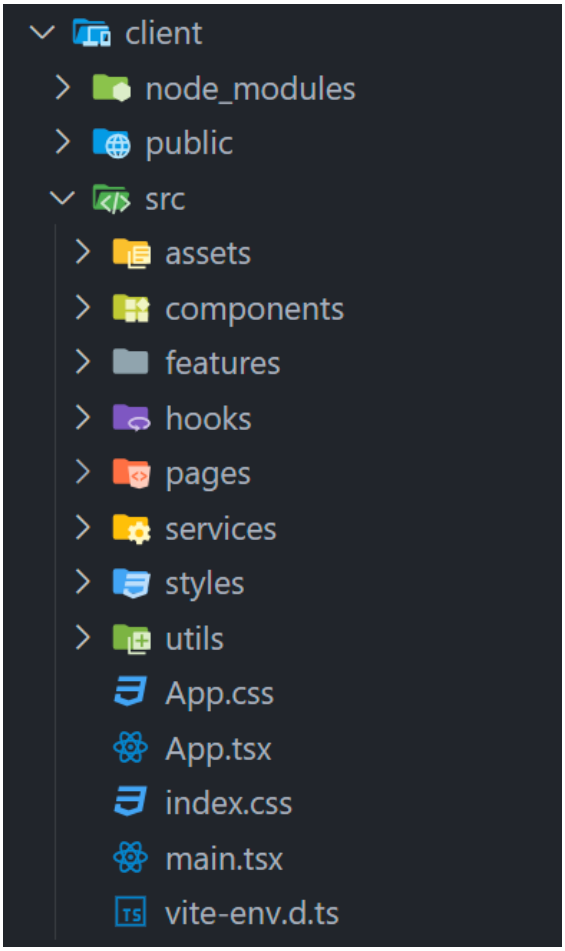
Predict for real-time use

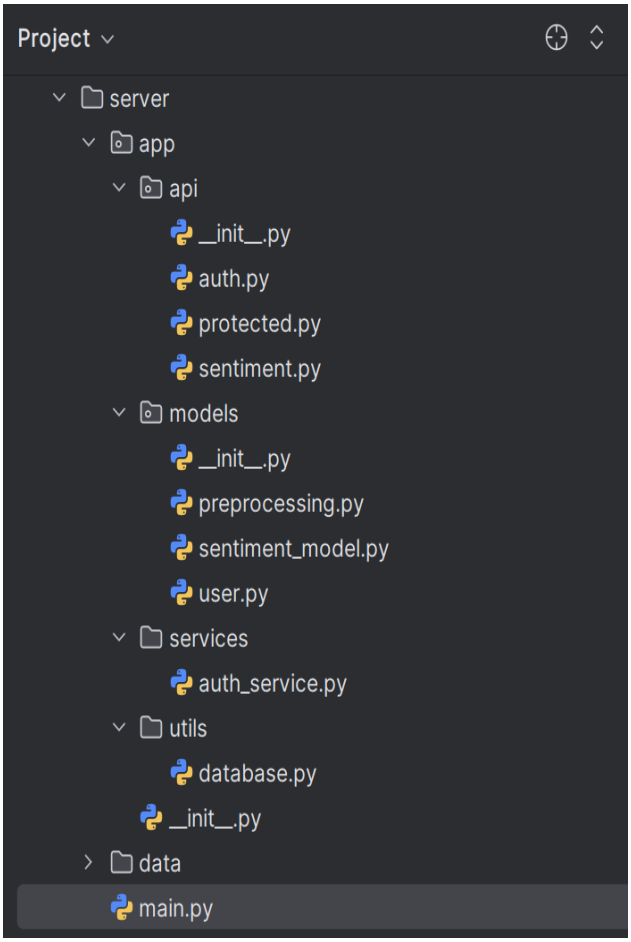
[11]:
```
# Predict with Probabilities for Real-Time Use.
custom_input = ["I'm feeling overwhelmed."]
custom_input_preprocessed = [preprocess_text(sentence) for sentence in custom_input]
custom_input_transformed = vectorizer.transform(custom_input_preprocessed)
custom_prediction = best_model.predict(custom_input_transformed)
custom_probabilities = best_model.predict_proba(custom_input_transformed)

predicted_label = label_encoder.inverse_transform(custom_prediction)
print("\nPredicted Class for the User Input:", predicted_label)
print("\nConfidence Scores for Each Class:", custom_probabilities)
```

```
Predicted Class for the User Input: ['Anxiety']

Confidence Scores for Each Class: [[3.64426242e-01 4.88491776e-03 4.81330080e-02 9.73443731e-02
  4.49041501e-03 1.90853826e-01 2.89820173e-01 4.70453513e-05]]
```

[12]:
```
import joblib
```

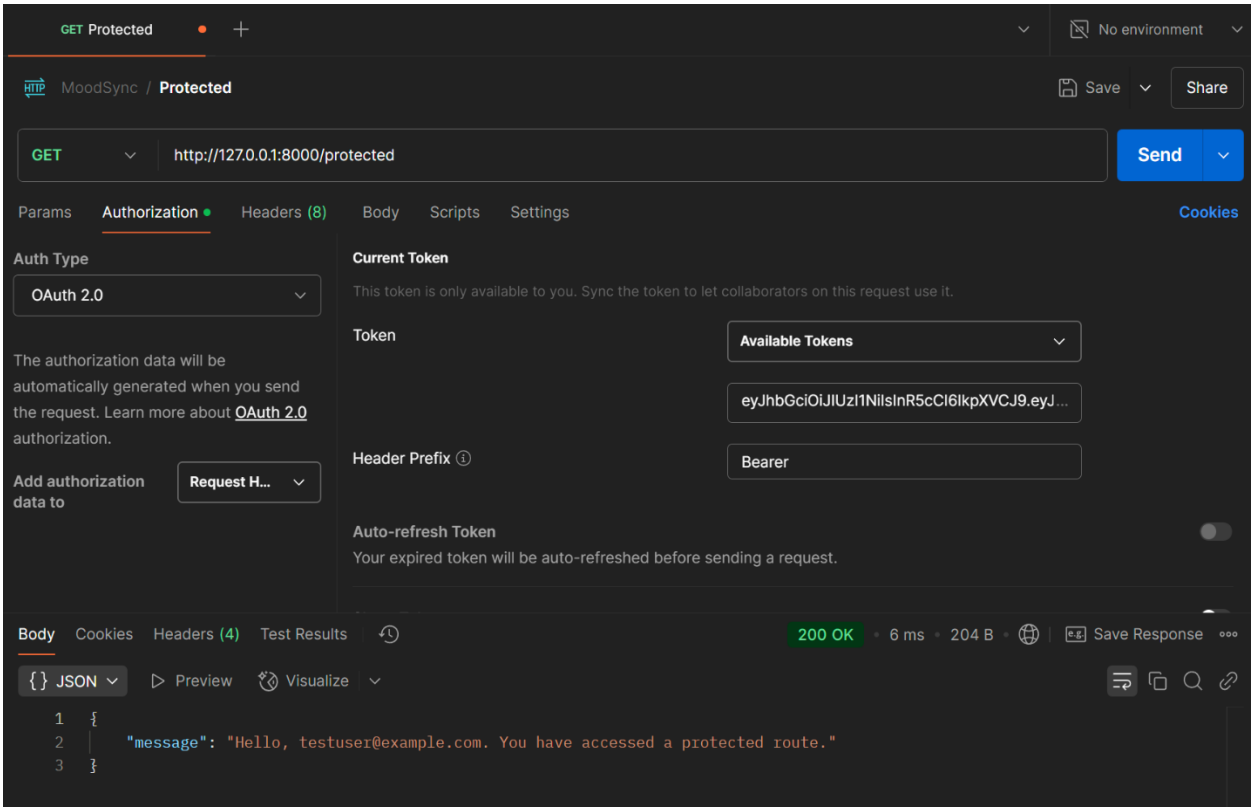
Code Snippets

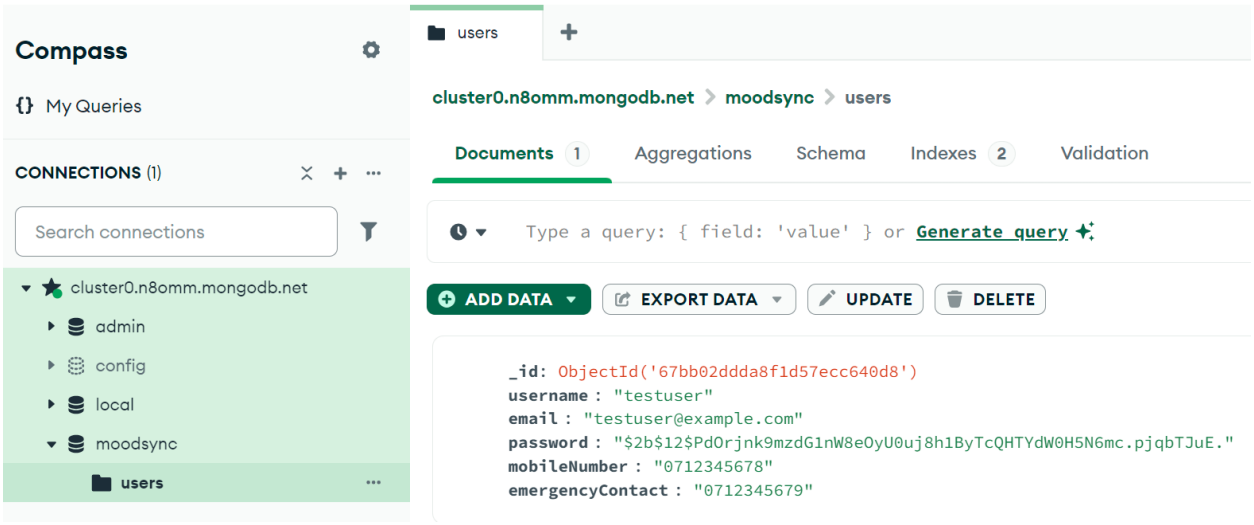




auth_service.py

```python
# Load environment variables
SECRET_KEY = os.getenv("SECRET_KEY")
ALGORITHM = os.getenv("ALGORITHM", "HS256")
ACCESS_TOKEN_EXPIRE_MINUTES = int(os.getenv("ACCESS_TOKEN_EXPIRE_MINUTES", "60"))

# Password hashing context
pwd_context = CryptContext(schemes=["bcrypt"], deprecated="auto")


# Utility functions for password handling
def hash_password(password: str) -> str:  1 usage  Gagana Methmal (GM)
    return pwd_context.hash(password)


def verify_password(plain_password: str, hashed_password: str) -> bool:  1 usage  Gagana Methmal (GM)
    return pwd_context.verify(plain_password, hashed_password)


# JWT Token creation
def create_access_token(data: dict, expires_delta: Optional[datetime.timedelta] = None) -> str:  1 usage  Gagana Methmal (GM)
    to_encode = data.copy()
    expire = datetime.datetime.utcnow() + (expires_delta or datetime.timedelta(minutes=ACCESS_TOKEN_EXPIRE_MINUTES))
    to_encode.update({"exp": expire})
    return jwt.encode(to_encode, SECRET_KEY, algorithm=ALGORITHM)


# Register new user
def register_user(user: UserIn) -> UserOut:  2 usages  Gagana Methmal (GM)
    if users_collection.find_one({"email": user.email}):
        raise HTTPException(
            status_code=status.HTTP_400_BAD_REQUEST,
            detail="Email already registered",
        )
```




Compass
My Queries
CONNECTIONS (1)
Search connections
cluster0.n8omm.mongodb.net
admin
config
local
moodsync
users
users
cluster0.n8omm.mongodb.net > moodsync > users
Documents 1
Aggregations
Schema
Indexes 2
Validation
Type a query: { field: 'value' } or Generate query
ADD DATA
EXPORT DATA
UPDATE
DELETE
_id: ObjectId('67bb02ddda8f1d57ecc640d8')
username : "testuser"
email : "testuser@example.com"
password : "$2b$12$PdOrjnk9mzdG1nW8eOyU0uj8h1ByTcQHTYdW0H5N6mc.pjqbTJuE."
mobileNumber : "0712345678"
emergencyContact : "0712345679"