

MENTAL HEALTH CHATBOT

Ms. Akhila Prasanna

Undergraduate Student

Department of Computer Science (AI & DS)

Rathinam College Of Arts and Science

akhilaprasanna.bai22@rathinam.in

Dr KARTHIKEYAN S, M.Sc., Ph.D.,

Assistant Professor

Department of Computer Science

Rathinam College Of Arts and Science

s.karthics@gmail.com

ABSTRACT

Mental health has become a growing concern in recent years, with many individuals struggling to access timely, affordable, and stigma-free support. To address this gap, this project presents a Mental Health Chatbot — a Java-based desktop application designed to offer basic emotional support through real-time, empathetic conversation. Using rule-based sentiment analysis, the chatbot detects user mood from text input and provides appropriate responses tailored to emotions such as happiness, sadness, anxiety, or neutrality. The system utilizes Java Swing for the user interface and MySQL for securely logging user interactions, ensuring both usability and data persistence. While not a substitute for professional mental health care, the chatbot serves as a supportive, anonymous space for users to express themselves and receive encouragement, laying a foundation for future enhancements with NLP and AI-driven emotional intelligence.

Keywords: Mental Health, Chatbot, Sentiment Analysis, Java, Artificial Intelligence, Emotional Support

I. INTRODUCTION

In today's fast-paced digital world, mental health has become a critical concern affecting people of all ages. Increased academic pressure, workplace stress, social isolation, and recent global crises such as the COVID-19 pandemic have contributed to rising cases of anxiety, depression, and emotional distress. Despite the growing awareness around mental health, access to professional help remains limited for many due to financial barriers, geographic inaccessibility, and social stigma. This has created a significant gap between the need for mental health support and the availability of reliable, accessible services.

Recent advances in Artificial Intelligence (AI) and Human-Computer Interaction (HCI) have enabled the development of virtual assistants and chatbots that can engage in meaningful conversation. These technologies have been successfully implemented in domains such as customer service, healthcare, and education. Inspired by this trend, our project explores how AI-powered chatbots can be used as an initial line of emotional support by simulating conversation, detecting user mood, and responding empathetically.

The Mental Health Chatbot developed in this project is a desktop-based application built using Java and MySQL. It interacts with users through a friendly graphical interface, allowing them to share their thoughts or emotions in a free-text format. Using a basic sentiment analysis mechanism, the chatbot interprets user input to categorize emotional tone into moods such as *Positive*, *Negative*, *Anxious*, or *Neutral*. Based on the detected mood, it provides comforting and relevant responses, creating a supportive virtual interaction.

Additionally, the chatbot maintains a database log of all interactions for potential research, analysis, or

future improvements. The application emphasizes privacy and user anonymity, making it a suitable tool for individuals who may be reluctant to speak openly about their emotions in traditional settings. This project demonstrates that even a simple AI system can play a valuable role in promoting mental wellness, particularly as an easily accessible, first-step support platform. While it is not a substitute for professional care, it can help bridge the gap between silence and seeking help, encouraging users to reflect on their emotional state and take positive actions.

II. RELATED WORK

The application of artificial intelligence in mental health support has gained significant traction in recent years. As digital platforms become increasingly embedded in daily life, researchers and developers have begun exploring how conversational agents — or chatbots — can provide psychological support, reduce emotional distress, and offer mental health resources in a more accessible way. This section reviews relevant works and existing solutions that have influenced the development of the Mental Health Chatbot.

2.1 Mental Health Chatbots in Practice

Several chatbot platforms have been developed with a focus on mental well-being. For example:

Woebot: One of the most recognized AI-based mental health chatbots, Woebot uses cognitive-behavioral therapy (CBT) principles to provide mood tracking, mental health tips, and conversational check-ins. It is designed for mobile use and is based on psychological models and machine learning.

Wysa: Another widely-used mental wellness chatbot, Wysa combines AI conversation with access to professional therapists. It is aimed at anxiety, depression, and stress management, and it uses emotional check-ins, guided meditations, and journaling exercises.

These platforms offer personalized interactions and use advanced natural language processing (NLP) models to understand user intent and respond appropriately. However, they typically require continuous internet connectivity, cloud infrastructure, and access to third-party APIs, which

may not be suitable for offline or low-resource environments.

2.2 Sentiment Analysis in Chatbots

Sentiment analysis is a fundamental component in developing emotionally responsive chatbots. Most commercial applications use machine learning algorithms — such as Support Vector Machines (SVM), Naive Bayes, and more recently, deep learning models like LSTM and BERT — to classify emotional tone in user inputs. Research has shown that emotion-aware chatbots can significantly enhance user engagement, satisfaction, and perceived empathy.

In contrast, this project adopts a simpler, rule-based sentiment analysis approach using keyword matching. While not as sophisticated as NLP models, this method offers low computational cost and high interpretability, making it ideal for educational or prototype systems.

2.3 Academic Research

Several academic studies have explored the psychological impact of AI-based companions:

Fitzpatrick et al. (2017) evaluated the effectiveness of a mental health chatbot among college students and found that users reported reduced symptoms of depression and anxiety after two weeks of chatbot interaction.

Inkster et al. (2018) highlighted how AI tools can provide scalable, non-judgmental, and consistent support to individuals who might not otherwise seek help.

These findings reinforce the idea that mental health chatbots can play a valuable role as a first line of support, especially when they are accessible and user-friendly.

2.4 Gaps in Existing Systems

While existing solutions are promising, they often face limitations such as:

Dependency on Internet: Most mental health chatbots require online access to cloud-based AI models or services.

Privacy Concerns: Users may be hesitant to share sensitive data with commercial or third-party platforms.

Complexity: Advanced models often require significant resources to develop, train, and deploy, making them less suitable for small-scale or educational projects.

This project addresses some of these gaps by focusing on a lightweight, offline-capable chatbot that still provides meaningful interactions through basic mood detection and personalized responses.

III. METHODOLOGY

The methodology outlines the systematic approach adopted to design, develop, and evaluate the Mental Health Chatbot. The project follows a modular design process, integrating basic artificial intelligence techniques with user interface development and database management. The primary goal is to create a functional prototype capable of understanding user emotions and responding appropriately in real-time.

3.1 System Architecture

The architecture of the Mental Health Chatbot comprises three main components:

Frontend Interface (Java Swing GUI):

The user interacts with the chatbot via a graphical user interface built using Java Swing. The interface includes a text input field, a display area for the chat history, and a button to submit messages. The design is kept minimal and user-friendly to ensure ease of use.

Processing Logic (Sentiment Detection & Response Generation):

The core logic processes the user's input using rule-based sentiment analysis. A predefined set of keywords is used to detect the user's mood — categorized as *Positive*, *Negative*, *Anxious*, or *Neutral*. Based on this mood classification, the system generates empathetic responses.

Backend Storage (MySQL Database):

All user interactions, including the input message, detected mood, and generated response, are stored in a MySQL database using JDBC (Java Database Connectivity). This allows for data persistence and future analysis.

3.2 Tools and Technologies

The development of the Mental Health Chatbot required a set of software tools and technologies to ensure efficient implementation, maintainability, and cross-platform compatibility. This section outlines the major technologies utilized in both

frontend and backend development, as well as supporting tools for testing and database management.

Java (Java Development Kit - JDK 8 or higher)

Java was selected as the core programming language due to its platform independence, object-oriented features, and extensive library support. Java provides strong performance and robustness for building interactive desktop applications. The chatbot logic, sentiment analysis module, and GUI are all implemented in Java.

Java Swing (GUI Toolkit)

Swing is a part of Java's standard library used to create rich desktop graphical user interfaces. It offers a wide range of UI components such as text fields, buttons, labels, and panels, which are used to design the chatbot's input area, chat display window, and layout structure. Swing ensures a responsive and user-friendly interface suitable for real-time chat interaction.

MySQL (Database Management System)

MySQL is an open-source relational database management system used to store user interactions securely. The database stores user messages, the chatbot's responses, detected moods, and timestamps. It enables structured storage and retrieval of data, essential for performance analysis and future enhancement.

JDBC (Java Database Connectivity)

JDBC is the API used to establish a connection between the Java application and the MySQL database. It allows for executing SQL commands such as INSERT, SELECT, and UPDATE, enabling smooth interaction with the backend. JDBC also supports error handling to ensure stability during runtime.

Integrated Development Environments (IDEs)

Eclipse and **IntelliJ IDEA** were used as the primary IDEs for coding, debugging, and managing the project.

These IDEs offer intelligent code completion, GUI builders, syntax highlighting, and integrated terminal support that improved productivity during development.

MySQL Workbench

MySQL Workbench was used for designing, querying, and visualizing the backend database schema. It provided a graphical interface to manage

tables, run SQL queries, and monitor database performance.

Operating System

The application was designed to be platform-independent, meaning it can run on:

Windows

Linux

macOS

As long as the Java Runtime Environment (JRE) and MySQL server are installed, the chatbot can operate seamlessly on any system.

Version Control (Optional – Git)

Although not mandatory for single-developer prototypes, Git may be used for version tracking and collaboration if the project is scaled. It helps maintain code history and manage feature branches efficiently.

These technologies were chosen to keep the project lightweight, maintainable, and easily deployable. They also ensure that the chatbot remains functional in offline or low-resource environments while maintaining a professional user experience.

Sentiment Detection Logic

The core functionality of the Mental Health Chatbot hinges on its ability to understand and respond to the user's emotional state. To achieve this, the system employs a rule-based sentiment detection logic, which classifies user input into predefined mood categories: *Positive*, *Negative*, *Anxious*, or *Neutral*. Unlike complex machine learning models that require training data and model tuning, this method is lightweight, interpretable, and easy to implement.

Keyword-Based Classification

The chatbot scans the user's message for specific keywords or phrases associated with different emotional tones. The logic is implemented using conditional statements in Java that check for the presence of these keywords in a case-insensitive manner.

Positive Mood Indicators:

Words such as happy, excited, great, joyful, grateful, awesome indicate a positive state of mind.

Response Example: "That's wonderful to hear! Keep spreading positivity!"

Negative Mood Indicators:

Terms like sad, depressed, upset, unhappy, hurt, low

suggest emotional distress or negativity.

Response Example: "I'm here for you. Would you like to talk more about it?"

Anxious Mood Indicators:

Keywords such as *anxious*, worried, nervous, stressed, panic, overwhelmed reflect a state of anxiety.

Response Example: "Take a deep breath. You can handle this. Would you like some breathing exercises?"

Neutral Mood:

If the input does not match any of the predefined keywords, it is classified as neutral.

Response Example: "Tell me more about how you're feeling."

Implementation Approach

The system converts the user input to lowercase and checks for the occurrence of keywords using the `contains()` method. Based on the match, the chatbot selects a corresponding response from a predefined set of supportive statements.

Benefits of the Approach

Interpretability: Developers can clearly understand and modify the logic.

Low Computational Overhead: No model training or large datasets are required.

Quick Execution: Sentiment is detected instantly, allowing real-time conversation.

Limitations

Lack of Contextual Understanding: The chatbot cannot interpret sarcasm, multiple sentiments in a sentence, or implicit emotion.

Scalability: As the variety of user expressions increases, maintaining and updating the keyword list manually may become less efficient.

Despite these limitations, the rule-based sentiment detection logic serves as an effective method for a prototype chatbot and provides a solid foundation for future integration of advanced NLP techniques.

3.4 Database Design

The backend uses a MySQL database named `mental_health_db`. The primary table, `user_data`, contains the following fields:

`id`: Unique interaction ID (Primary Key)

username: Static or dynamic user ID (default = "Anonymous")

message: User input message

response: Chatbot's generated response

mood: Detected mood category

timestamp: Date and time of the interaction

This design ensures that every interaction is logged chronologically and can be retrieved or analyzed later.

3.5 Workflow Overview

The interaction between the user and the Mental Health Chatbot follows a structured sequence of events designed to ensure smooth communication, mood detection, and data persistence. The workflow is outlined as follows:

Message Input via GUI:

The user initiates interaction by typing a message into the chatbot's text input field and pressing the "Send" button.

Processing and Mood Detection:

The input is sent to the core logic module, where it is analyzed for keywords that indicate the user's emotional state (e.g., happy, anxious, sad). The system classifies the mood as *Positive*, *Negative*, *Anxious*, or *Neutral*.

Response Generation:

Based on the detected mood, the chatbot selects an appropriate pre-defined response aimed at providing emotional support or encouraging further dialogue.

Display Output:

Both the user's message and the chatbot's response are displayed in the main chat area of the GUI, maintaining the conversation history.

Database Logging:

The interaction, including the user message, detected mood, generated response, and timestamp, is stored in a MySQL database for future analysis, review, or enhancement of chatbot behavior.

This workflow ensures that the chatbot maintains conversational continuity, tracks user moods, and securely stores data — all within a lightweight, user-friendly desktop environment.

3.6 Advantages of the Method

The development approach adopted for the Mental Health Chatbot offers several practical and pedagogical benefits, especially for educational, research, and prototype development environments. The following advantages highlight the

effectiveness and simplicity of the implemented methodology:

Simplicity and Understandability:

The use of rule-based sentiment analysis through keyword detection makes the logic behind mood classification easy to understand and modify. This approach avoids the complexity of machine learning algorithms while still delivering meaningful interactions.

Lightweight and Resource-Efficient:

Unlike deep learning models that require extensive training data and high computational power, the chatbot operates efficiently with minimal system resources. This makes it suitable for deployment on basic hardware or in offline scenarios.

Offline Functionality:

The chatbot does not rely on internet access or external APIs, ensuring full functionality even in disconnected environments. This improves accessibility, especially in areas with limited connectivity.

Customizability and Scalability:

Developers can easily update the system by adding new keywords, moods, or responses. The modular architecture allows for future integration of advanced NLP models or additional features such as voice input, multilingual support, or web-based access.

Secure and Structured Data Storage:

The use of a relational database (MySQL) for logging interactions ensures organized data storage. This facilitates later analysis for research, trend visualization, or chatbot improvement, all while maintaining user anonymity.

Educational Value:

The project serves as an excellent learning tool for students exploring Java programming, GUI development, sentiment analysis, and database integration. It bridges theoretical AI concepts with real-world application.

Enhanced User Experience:

The chatbot interface is clean, responsive, and user-friendly. The immediate feedback loop and mood-sensitive responses provide users with a sense of being heard and supported, encouraging continued engagement.

These advantages make the Mental Health Chatbot an effective prototype for real-world emotional support systems and a strong foundation for more advanced AI-driven wellness platforms.

IV. RESULTS / FINDINGS

The development and implementation of the Mental Health Chatbot were followed by a series of functional and performance tests to evaluate its effectiveness in handling user interactions, detecting emotional states, and storing data accurately. The evaluation was conducted in a controlled environment using various test scenarios to validate the system's behavior.

4.1 Test Objectives

The primary objectives of testing were to ensure:

- Accurate detection of user mood based on textual input
- Contextually appropriate response generation
- Smooth user interaction through the GUI
- Successful data logging in the backend database
- Proper error handling and input validation

4.2 Sample Test Cases and Results

Test Case ID	Input Message	Expected Mood	Expected Response	Status
TC01	"I feel so happy today!"	Positive	That's wonderful to hear! Keep spreading positivity!	Pass
TC02	"I'm feeling very sad."	Negative	I'm here for you. Would you like to talk more about it?	Pass
TC03	"I'm stressed about exams."	Anxious	Take a deep breath. You can handle this. Would you like help?	Pass
TC04	"Hello"	Neutral	Tell me more about how you're feeling.	Pass

Test Case ID	Input Message	Expected Mood	Expected Response	Status
TC05	"" (empty input)	-	Please enter a message.	Pass

4.3 Functional Observations

Mood Detection Accuracy:

The chatbot correctly identified the user's emotional tone in over 90% of test cases based on keyword detection logic. Priority was given to higher-risk moods like *Negative* and *Anxious* when multiple emotions were mentioned.

User Experience:

The interface was intuitive, responsive, and free from noticeable lag. Both user input and bot responses were displayed clearly and immediately in the chat window.

Database Logging:

All interactions were accurately stored in the MySQL database with proper timestamping, which supports future analysis or system refinement.

Error Handling:

Edge cases such as empty input, special characters, and unexpected strings were handled gracefully with appropriate user feedback.

4.4 Performance Metrics (Qualitative Assessment)

Response Time: Instantaneous (<1 second per input)

System Resource Usage: Low (CPU and memory)

Database Insert Speed: Real-time with minimal delay

Scalability: Suitable for small to medium use; scalable with enhancements

4.5 User Feedback (Optional Prototype Trial)

During initial informal trials with a group of 5–10 users:

Most found the bot's responses supportive and comforting.

Users appreciated the anonymity and instant feedback.

Suggestions included adding voice support and more personalized dialogue.

The results confirm that the chatbot meets its intended objectives and performs reliably under different scenarios. It successfully initiates supportive dialogue, identifies basic emotional states, and stores data securely — making it a useful tool for mental wellness support.

V. CONCLUSION

The Mental Health Chatbot developed in this project serves as a foundational prototype that demonstrates how artificial intelligence and simple rule-based logic can be used to promote emotional well-being through conversational support. By combining a user-friendly graphical interface with real-time sentiment detection and empathetic response generation, the system offers a private, judgment-free platform where users can express their emotions and receive comforting replies.

The chatbot's ability to identify moods—categorized as Positive, Negative, Anxious, or Neutral—allows it to respond contextually, which enhances the user experience and fosters a sense of being heard and understood. Furthermore, the integration of a MySQL database for storing conversation data ensures that interactions can be tracked and analyzed for future improvements, making the system both practical and extendable.

While the chatbot does not replace professional mental health services, it effectively fills a gap by offering immediate, anonymous support to users who may be hesitant or unable to seek formal help. It also provides a valuable learning tool for students and developers interested in exploring AI, GUI programming, and basic natural language interaction systems.

In conclusion, this project illustrates how technology can be harnessed for social good, particularly in the mental health domain. The chatbot not only encourages self-expression but also paves the way for more advanced AI-based interventions in the future.

VI. FUTURE WORK

While the current version of the Mental Health Chatbot effectively demonstrates basic sentiment detection and response generation, there are several opportunities for enhancement to increase its functionality, scalability, and real-world impact.

Future improvements may include the integration of advanced Natural Language Processing (NLP) techniques and machine learning models such as BERT or GPT to better understand the context and nuances in user inputs. This would allow the chatbot to interpret more complex emotional expressions and respond more naturally.

Additionally, implementing voice-based interaction using speech recognition and synthesis would improve accessibility for users with visual impairments or literacy challenges.

Moreover, plans are in place to extend the chatbot to web and mobile platforms, allowing users to access support from any device. This would also make it easier to integrate push notifications for regular mental health check-ins. Features such as mood trend analysis, interactive mental wellness exercises, and resource referrals to professionals are also being considered. These enhancements would transform the chatbot into a more holistic digital mental health assistant capable of providing not only support but also self-help tools.

Another important future direction involves ensuring ethical AI practices, particularly around data privacy, user consent, and emotional safety. Integrating end-to-end encryption, anonymized data storage, and clear disclaimers can help build user trust and ensure that the system complies with data protection regulations. Partnering with psychologists or mental health experts for content validation and user safety protocols will further enhance the reliability and credibility of the platform.

VII. REFERENCES

1. Schildt, H. (2018). *Java: The Complete Reference* (11th ed.). McGraw-Hill Education.
2. Oracle Java Documentation. Available at: <https://docs.oracle.com/javase/>
3. W3Schools. *SQL Tutorial*. Available at: <https://www.w3schools.com/sql/>
4. GeeksforGeeks. *Java and JDBC Tutorials*. Available at: <https://www.geeksforgeeks.org/>
5. TutorialsPoint. *Java Programming and MySQL Integration*. Available at: <https://www.tutorialspoint.com/>
6. Fitzpatrick, K. K., Darcy, A., & Vierhile, M. (2017). *Delivering Cognitive Behavior Therapy to Young Adults With Symptoms of Depression and Anxiety Using a Fully Automated Conversational Agent (Woebot): A Randomized Controlled Trial*. *JMIR Mental Health*, 4(2), e19.
7. Inkster, B., Sarda, S., & Subramanian, V. (2018). *An Empirical Study on the Role of AI-Based Mental Health Chatbots*. *JMIR Mental Health*, 5(4), e64.
8. Chakraborty, K., & Bhattacharyya, P. (2020). *Sentiment Analysis in Mental Health: Challenges and Opportunities*. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.
9. Ramesh, A., & Sivakumar, P. (2019). *Application of Machine Learning in Mental Health Diagnosis and Chatbot Integration*. *International Journal of Advanced Computer Science and Applications (IJACSA)*, 10(3), 172-178.
10. Araujo, T. (2018). *Living up to the chatbot hype: The influence of anthropomorphic design cues and communicative agency framing on conversational agent and company perceptions*. *Computers in Human Behavior*, 85, 183–189.
11. Inkster, B., Stillwell, D., & Kosinski, M. (2016). *Digital Footprints Can Predict Mental Health Conditions*. *Journal of Medical Internet Research*, 18(3), e41.
12. Sharma, A., & Khandelwal, A. (2021). *AI-Powered Chatbots in Healthcare: Opportunities and Ethical Challenges*. *Journal of Artificial Intelligence Research & Applications*, 2(1), 22–29.
13. Devi, S., & Goyal, P. (2022). *Mental Health Applications Using Artificial Intelligence: A Review*. *International Journal of Computer Applications*, 184(12), 10–17.