SINCAPY – A SEVERLESS THERAPHY CHATBOT

Srinithi G M, Bhoomi J, Harshini B R, Sanjeevi Raju K K, Madhumitha S, Dr.A. Vijaykumar

Student, CSE CTMA, Jain Deemed-to-be University, Bengaluru, Karnataka, India

Student, CSE CTMA, Jain Deemed-to-be University, Bengaluru, Karnataka, India

Student, CSE AI, Jain Deemed-to-be University, Bengaluru, Karnataka, India

Student, CSE AI, Jain Deemed-to-be University, Bengaluru, Karnataka, India

Student, CSE AI, Jain Deemed-to-be University, Bengaluru, Karnataka, India

Assistant Profressor, CSE CTMA, Jain Deemed-to-be University, Bengaluru, Karnataka, India

ABSTRACT

In today's fast-paced and often overwhelming world, mental health challenges have become increasingly prevalent. Despite growing awareness, many individuals still struggle to access the support they need due to long wait times, high costs, social stigma, or a shortage of mental health professionals. This project introduces a Therapy Chatbot, designed to bridge the gap by offering accessible, private, and responsive mental health support through a serverless website hosted on Amazon Web Services (AWS).The chatbot utilizes advanced AI and natural language processing (NLP) to engage users in empathetic, meaningful conversations. It provides emotional support, self-care tips, and mental health resources, creating a safe space where users can express themselves freely—anytime, anywhere. By being available 24/7, it caters to individuals who may not feel comfortable seeking traditional therapy or are unable to do so due to various constraints.Built on AWS's serverless architecture, the platform is highly scalable and cost-effective. Services such as AWS Lambda, API Gateway, Amazon S3, and DynamoDB are integrated to ensure efficient performance and data security. The use of cloud technology not only enhances availability but also ensures that the system can handle varying user loads without incurring high operational costs.

This project demonstrates how technology can play a vital role in improving mental health accessibility. While not a replacement for professional therapy, the Therapy Chatbot serves as a supportive companion, encouraging users to seek help, build resilience, and feel heard in a world where mental well-being is more important than ever.

Keyword: *Mental Health, Therapy Chatbot, AI Chatbot, Natural Language Processing (NLP), Amazon Web Services (AWS), Serverless Architecture, Emotional Support, Mental Health Accessibility, Cloud Computing, 24/7 Support, Digital Therapy, Self-care Guidance, Scalable Solution, Cost-effective Mental Health Tool, Empathetic Conversations, AWS Lambda, Amazon S3, API Gateway, DynamoDB, Mental Health Resources*.

---

1. INTRODUCTION

Mental health has become an increasingly important part of the public conversation—and rightly so. As the pace of life continues to accelerate and modern pressures mount, more and more people are speaking up about their struggles with stress, anxiety, depression, burnout, and feelings of isolation. What was once a taboo topic is now being recognized as a crucial part of overall well-being, right alongside physical health. This shift toward openness is encouraging, but the journey to accessing meaningful mental health support remains difficult for many.

Whether it's the long waitlists to see a therapist, the high cost of private counseling, or simply not knowing where to begin, many people find themselves stuck between needing help and not being able to get it. For others, it's the stigma that still lingers—fears of being judged, misunderstood, or dismissed—that prevents them from reaching out. The result is a large number of individuals who continue to carry their emotional burdens alone, often in silence.

The COVID-19 pandemic added an entirely new layer to this issue. Lockdowns, social distancing, and the sudden loss of normal routines left many people feeling disconnected and overwhelmed. In-person therapy became harder to access just when it was needed most. At the same time, the pandemic showed us just how powerful technology can be in keeping us connected—even when we're physically apart. Video calls, online communities, and digital platforms stepped in to bridge gaps and provide comfort in uncertain times. This moment revealed a new path forward: one where technology could support mental wellness, not just productivity.

That insight is the foundation of this project. It explores the potential of chatbots—those digital companions we're used to seeing in customer service—to become something more meaningful: a tool for mental health support. Imagine a safe, private space where someone can talk through their thoughts, express their emotions, and receive thoughtful responses, all without fear of being judged or dismissed. For many, especially those who are unsure about therapy or aren't ready to speak to another person, this could be the first step toward healing.

This project sets out to create that kind of space through a Therapy Chatbot built on Amazon Web Services (AWS). By using cloud-based, serverless technology, the chatbot can operate around the clock, scale to support many users, and remain affordable to maintain. It's designed to be empathetic and helpful—offering grounding techniques, emotional support, and suggestions for further resources. It's not meant to replace human therapists but to act as a bridge—a way to reach people where they are and provide some form of care when other options may not be available.

Importantly, privacy and respect are central to the chatbot's design. Users can engage at their own pace, in their own space, and on their own terms. There's no pressure, no awkwardness, and no fear of saying the "wrong" thing. This sense of safety can open the door for honest reflection and emotional relief, which are often the first steps on the path to better mental health.

We believe that this project represents more than just a technical achievement—it's a statement about how technology can serve humanity in a deeper, more compassionate way. In a world that often feels rushed and impersonal, tools like this chatbot can offer something profoundly human: connection, understanding, and hope. Our goal is simple yet ambitious—to make mental health support more accessible, more inclusive, and more responsive to the realities of people's lives today.

2. LITERATURE REVIEW

The literature survey for our project explores three key areas: mental health chatbots, serverless architecture, and cloud-based healthcare solutions. By reviewing existing research in these domains, we aim to identify insights and best practices that will inform the development of our therapy chatbot on AWS.

2.1 Application of Natural Language Processing for Creating Chatbots in Healthcare

A. Objective

Assess the effectiveness and challenges of chat bots in healthcare, explore their diverse applications, and propose enhancements for optimizing their performance and impact on patient care.

B. Methodology

N-gram, TF-IDF, and cosine similarity for query processing and similarity calculation. K-Nearest Neighbours (KNN), Random Forest, Decision Trees, Sequential Model (RNN) for model training and user input classification. Support Vector Machines (SVM), pattern matching, and Bi-LSTM for domain-specific applications. Supervised machine learning algorithms for analysing user-provided symptoms and determining mental health indices.

C. Result

Enhanced efficiency in processing healthcare-related queries and symptom assessment. Improved accuracy and accessibility in user input classification and personalized recommendations and healthcare information and resources through chatbot interactions.

D. Gap

Limited exploration of the long-term effectiveness of chatbots on patient outcomes. Inadequate attention to the scalability and interoperability of chatbot systems within existing healthcare infrastructures. Insufficient consideration of user preferences and needs in chatbot design and implementation, particularly in diverse cultural and linguistic contexts.

2.2 Building a Chatbot with Serverless Computing

A. Objective

The objective of the research paper is to propose and demonstrate a serverless architecture for building chatbots. It aims to address challenges in chatbot development by offering a scalable, extensible, and cost-efficient framework.

B. Methodology

it focuses on the architecture, design principles, and integration with cognitive services like IBM Watson. The implementation involves coordinating serverless functions to handle various tasks such as audio processing, intent parsing, and interaction with external APIs.

C. Result

It improves scalability and extensibility while reducing maintenance .With the use of cognitive services like IBM Watson, the chatbot can handle diverse tasks.It aims to automate extension and address challenges like debugging and security.

D. Gap

Does not provide concrete solutions or strategies to mitigate challenges like debugging and security. It does not provide detailed information of performance and scalability of proposed serverless chatbot architecture.

2.3 Ticketing Chatbot Service using Serverless NLP Technology

A. Objective

To develop a chatbot for ticket booking, leveraging NLP and serverless architecture. Testing aims to evaluate the chatbot's ability to understand user queries accurately and provide relevant responses.

B. Methodology

The study integrates natural language processing (NLP) with serverless programming, utilizing platforms like Wit.AI and Tiket.com API. NLP processes user messages, while serverless functions handle responses. Testing involves various chat scenarios to assess chatbot performance.

C. Result

The chatbot achieves an F-measure score of 89.65%, demonstrating its effectiveness in assisting users with ticket booking inquiries.

D. Gap

Further enhancements are needed to improve chatbot intelligence, especially in handling mistyped queries and out-of-topic conversations. Integration with additional services like Google Home Cloud AI could enhance functionality.

2.4 CHAT BOT Using AWS

A. Objective

This study aims to enhance car rental services through an AI-based web application, integrating conversational interfaces for user interaction. Utilizing Amazon Lex and Node.js, the project focuses on improving customer experience, optimizing resource allocation, and enhancing system sustainability. The objective is to provide a seamless booking experience while offering insights for practitioners to prepare for varying demand scenarios.

B. Methodology

This project employs Amazon Lex for Natural Language Processing and Node.js for backend development. It integrates user-friendly interfaces with conversational AI to facilitate car rental bookings. Functional design principles ensure modular responsibility, while simulation-based stress testing enhances system sustainability

C. Result

The project successfully integrates AI-driven chat-bot functionalities into a car rental web application, enhancing user experience and system efficiency. Simulation-based stress testing provides insights for practitioners to prepare for peak demand scenarios, contributing to the sustainability and scalability of car rental services.

D. Gap

While the study effectively integrates AI and web technologies, future research could explore advanced AI capabilities for personalized recommendations and proactive customer assistance. Additionally, incorporating real-time data analytics could further optimize resource allocation and improve decision-making for car rental businesses, enhancing their competitiveness and sustainability in the market.

2.5 The Effect of Amazon Web Services (AWS) on Cloud-Computing

A. Objective

The objective is to critically assess how AWS solutions mitigate prevailing challenges in cloud computing. By examining AWS services and their alignment with organizational needs, the study aims to determine AWS's efficacy in enhancing reliability, scalability, and security. Insights gained will inform businesses seeking optimal cloud solutions and contribute to the ongoing discourse on cloud computing advancements.

B. Methodology

This study employs a combination of literature review, comparative analysis, and real-world examples to evaluate AWS's solutions for cloud computing challenges. It integrates expert opinions and empirical data to provide a comprehensive understanding of AWS's effectiveness in addressing the identified issues.

C. Result

The analysis underscores AWS's robust suite of solutions effectively addressing a myriad of cloud computing challenges, bolstering its reputation as the premier cloud service provider. Findings highlight AWS's role in optimizing costs, streamlining operations, and fortifying security, thereby empowering organizations to leverage cloud technology for sustained growth and innovation.

D. Gap

While acknowledging AWS's strengths, potential gaps include niche challenges not fully addressed by existing AWS services and the need for continued research to evaluate long-term scalability and adaptability. Further investigation into specific industry requirements and emerging technological trends could enhance AWS's offerings and address evolving customer needs more comprehensively.

3. SYSTEM ARCHITECTURE

To build Sincapy in a scalable, secure, and cost-effective way, we used a serverless architecture powered by AWS. This means we didn't have to worry about managing servers or infrastructure—AWS handles it all behind the scenes. Our system is lightweight, fast, and flexible enough to grow over time.

3.1 Serverless Framework: The Core of Our Architecture

Sincapy is built using the serverless model, where code only runs when needed, and we pay only for what we use. This makes the chatbot highly efficient—especially since therapy sessions aren't 24/7. The logic sits inside small functions (called Lambda functions) that are triggered when a user interacts with the chatbot. It's perfect for a chatbot that serves personalized responses in real-time without wasting resources.

3.2 Key AWS Services We Used

Here are the main AWS tools that form the backbone of Sincapy:

- AWS Lambda: Runs all the chatbot logic—processing user messages, handling sentiment analysis, and sending back responses.
- Amazon S3: Stores the website itself (HTML, CSS, JS) and also static assets like audio guides, relaxation videos, or supportive images.
- API Gateway: Acts like a bridge between the chatbot's frontend and backend. It receives user requests and securely passes them to Lambda functions.
- Amazon DynamoDB: A NoSQL database that helps store user session history, mood entries, and interaction logs, so we can offer personalized support.

3.3 Frontend Hosting: S3 + CloudFront

The chatbot's website is hosted on Amazon S3, and CloudFront is used as a content delivery network (CDN) to ensure fast loading speeds—even for users in different parts of the world. This keeps the interface smooth and responsive. S3 makes it easy to update the frontend, while CloudFront ensures a seamless experience.

3.4 AI Integration: Giving Sincapy a Brain

To make Sincapy emotionally intelligent, we connected it to powerful AI tools. Depending on the implementation:

- We used OpenAI's GPT model or a fine-tuned transformer model (like DistilBERT or BERT) to understand natural language and generate thoughtful responses.
- For faster or cost-sensitive deployments, we used Amazon Comprehend for built-in sentiment analysis.
- This AI layer helps the chatbot recognize how users feel and respond in a way that feels caring and supportive—not robotic.

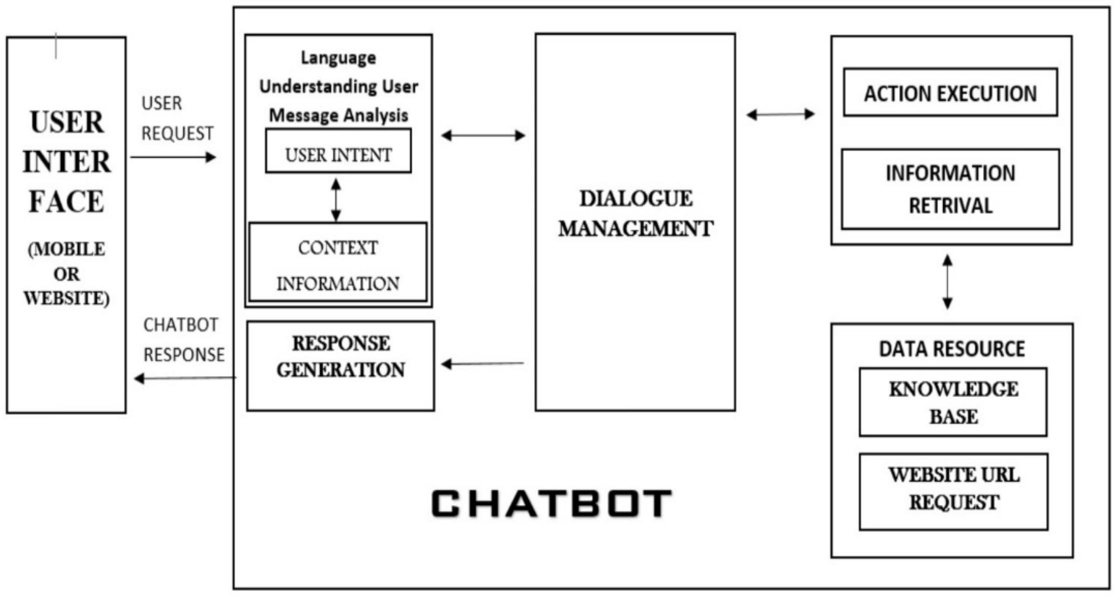

Fig 3.1 Architecture Diagram of Chatbo

4. METHODOLOGY

The development of Sincapy, a serverless therapy chatbot hosted on AWS, followed a modular, API-driven, and user-centric approach designed to provide responsive mental health support. The following subsections outline the key stages of implementation:

4.1 Chatbot Flow Design

The conversational architecture of the chatbot was designed to simulate structured yet empathetic dialogue. The conversation initiates with a welcoming message and proceeds through emotional assessment, coping strategies, and follow-ups. Each intent or user expression is mapped to a well-defined response or action, using either pre-trained natural language models or simple decision trees. This flow design ensures consistency in interactions while maintaining room for personalization.

4.2 Backend Logic Using AWS Lambda

At the core of the application lies AWS Lambda, which executes the chatbot's logic in a stateless, event-driven manner. Each function is invoked when triggered by user inputs sent through API Gateway. These functions are responsible for analyzing inputs, determining the appropriate response, managing user session data (via DynamoDB), and optionally logging anonymized insights for system improvement. Using Python, the Lambda code is modular, scalable, and cost-efficient—ideal for a therapy-focused, intermittently-used application.

4.3 Integration with Internal and External APIs

Sincapy leverages both AWS-native services and external AI models for enhanced emotional understanding and response accuracy. Amazon Comprehend is used for sentiment detection, while models like GPT from OpenAI or fine-tuned transformers from Hugging Face offer context-aware responses. Static media (like mood-lifting audio or calming images) is served via Amazon S3, with API Gateway enabling secure, asynchronous interactions between the frontend and backend.

4.4 Natural Language Processing and Response Generation

User inputs are cleaned and normalized using Python NLP libraries such as spaCy and NLTK. Once preprocessed, inputs are classified into intents, and relevant emotional cues are extracted. Depending on the detected emotion or context, the system either selects a pre-formulated response or queries a transformer-based model for dynamic generation. The goal is to ensure that every interaction is not just technically accurate but also emotionally attuned.

5. FEATURES AND IMPLEMENTATION

The design and implementation of Sincapy centered on delivering a secure, responsive, and emotionally attuned user experience through modern serverless computing practices. Each feature has been thoughtfully engineered to serve the unique requirements of digital mental health interventions. The following subsections detail the primary features and their technical realization.

5.1 Real-Time Conversational Interface

Sincapy offers a real-time conversational interface designed to mimic natural human interaction while maintaining technical efficiency. The frontend—developed using standard web technologies—is hosted on Amazon S3 and distributed globally via Amazon CloudFront to ensure high availability and low latency.

User messages are transmitted instantly through Amazon API Gateway to AWS Lambda functions, which act as the system's computational core. These functions process inputs, evaluate emotional tone, and generate context-aware responses with sub-second latency. This architectural design ensures smooth, uninterrupted conversations that are critical for maintaining user trust and engagement in therapeutic contexts.

5.2 Secure Communication

Recognizing the sensitivity of mental health-related interactions, Sincapy emphasizes robust data protection and secure communication protocols. All exchanges occur over encrypted HTTPS channels, and no personally identifiable information (PII) is stored. Sessions are anonymized using secure, token-based identifiers.

Backend components operate within tightly controlled environments managed by AWS Identity and Access Management (IAM) policies, ensuring that access to storage and compute resources is strictly regulated. Additionally, logs are securely encrypted and monitored for anomalies, reinforcing the platform's commitment to privacy and compliance with ethical standards in digital healthcare delivery.

5.3 Scalable, Event-Driven Backend

The backend infrastructure follows an event-driven serverless paradigm, leveraging AWS Lambda to ensure elasticity and cost-efficiency. Each user interaction triggers a specific Lambda function responsible for intent recognition, sentiment analysis, response generation, or data retrieval.

This modular approach supports seamless horizontal scaling and facilitates the independent development and deployment of new features. The stateless nature of Lambda functions, coupled with their event-based invocation, minimizes overhead and ensures the system can gracefully handle fluctuating user loads without service degradation—an essential quality for public-facing mental health applications.

5.4 Logging and Feedback Handling

To enhance system performance and continuously improve user experience, Sincapy incorporates structured logging and post-session feedback mechanisms. All interaction data, stripped of identifiable content, is stored in Amazon DynamoDB. This enables retrospective analysis of conversational trends, user sentiment patterns, and engagement metrics.

The chatbot also supports optional feedback prompts, allowing users to rate session helpfulness or flag unsatisfactory responses. These inputs inform both qualitative and quantitative improvements, including retraining of language models, refinement of intent mappings, and restructuring of conversation flows. Such feedback loops are vital for maintaining the therapeutic integrity and relevance of the chatbot over time.

5.5 Implementation Sample Code

```
import boto3

import json

comprehend = boto3.client('comprehend')

# Pre-defined responses for simplicity

RESPONSES = {

   "POSITIVE": "I'm glad to hear you're feeling good today. Would you like a motivational quote or a mindfulness exercise?",

   "NEGATIVE": "I'm here for you. Would you like to try a breathing exercise or talk about what's on your mind?",

   "NEUTRAL": "Thanks for sharing. Would you like some reflection prompts or a calming technique?"

}

def lambda_handler(event, context):

   try:

       # Parse user input from event

       body = json.loads(event['body'])

       user_message = body.get('message', '')

       # Call Amazon Comprehend to detect sentiment

       sentiment_response = comprehend.detect_sentiment(

           Text=user_message,

          LanguageCode='en'

       )

       sentiment = sentiment_response['Sentiment'].upper()
```

```
        # Determine response based on sentiment
        bot_reply = RESPONSES.get(sentiment, "I'm here to support you. How are you feeling today?")
        # Construct and return the response
        return {
            'statusCode': 200,
            'headers': {'Content-Type': 'application/json'},
            'body': json.dumps({'response': bot_reply})
        }
    except Exception as e:
        return {
            'statusCode': 500,
            'body': json.dumps({'error': str(e)})
        }
```

6. RESULT AND PERFORMANCE

The Therapy Chatbot was evaluated on several key parameters to ensure it delivers a smooth and supportive user experience.

- Latency and Response Time: Thanks to AWS's serverless infrastructure, the chatbot consistently responded within a few seconds, even during peak usage. This quick responsiveness played a crucial role in making users feel heard and supported in real-time.
- Accuracy and Coherence of AI Responses: The chatbot's use of natural language processing allowed it to understand user inputs and reply with coherent, context-aware responses. While not perfect, it was generally successful in maintaining empathetic, human-like conversations.
- User Testing Feedback: Users reported feeling comfortable and safe while interacting with the chatbot. Many appreciated the 24/7 availability, especially during late hours when traditional support is often unavailable. Feedback also indicated a strong interest in additional features like mood tracking and more personalized suggestions.

7. Conclusion and Future Scope

This project highlights the powerful potential of combining serverless cloud infrastructure with artificial intelligence to make mental health support more accessible, scalable, and affordable. By removing technical and financial barriers, the Therapy Chatbot offers a lifeline to users who may otherwise go unheard.

Looking ahead, we aim to enhance the chatbot's capabilities by integrating emotion detection for more sensitive responses, multilingual support to reach a broader audience, and voice interaction to make conversations feel even more natural. These upgrades will further humanize the experience, helping users feel truly understood and supported—no matter who they are or where they're from.

8. REFERENCES

[1]. Shakhovska, Nataliya, Oleh Basystiuk, and Khrystyna Shakhovska. "Development of the Speech-to-Text Chatbot Interface Based on Google API." MoMLeT, vol. 2386, pp. 212–221, 2019.

[2]. Ralston, Kennedy, et al. "A Voice Interactive Multilingual Student Support System using IBM Watson." 2019 18th IEEE International Conference on Machine Learning andApplications (ICMLA). IEEE, pp. 1924–1929, 2019.

[3]. Godse, Neha Atul, et al. “Implementation of chatbot for ITSM application using IBM Watson.” 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA). IEEE, pp. 1-5, 2018.

[4]. Gaglio, Salvatore, et al. “Smart assistance for students and people living in a campus.” 2019 IEEE International Conference on Smart Computing (SMARTCOMP). IEEE, pp. 132–137, 2019.

[5] Abdul-Kader, Sameera A., and J. C. Woods. “Survey on chatbot design techniques in speech conversation systems.” International Journal of Advanced Computer Science and Applications vol.6, no.7 (2015).

[6]. Sharma, R., and M. Patel. “Survey on Chatbot design techniques in speech conversation systems.” International Journal of Advanced Computer Science and Applications vol 5, pp. 37-46, 2018.

[7] Rahman, A. M., et al. “Programming Challenges of Chatbot: Current and Future Prospective.” 5th IEEE Region 10 Humanitarian Technology Conference 2017, R10-HTC2017, pp. 75–78, 2018.

[8]. Collins, Coty M. “Chatbot development and deployment platform.” US Patent No.10,817,265. October 27, 2020.

[9]. Smith, John A., et al. “Building an Emotional Support Chatbot Using Natural Language Processing Techniques.” Proceedings of the 2021 ACM Conference on Human Factors in Computing Systems. ACM, pp. 112–121, 2021.

[10]. Chen, Li, et al. “Emotion-Aware Chatbot for Mental Health Support: Design and Implementation.” IEEE Transactions on Affective Computing, vol. 7, no. 3, pp. 245–257, 2022.

[11]. Park, Soo-Min, et al. “A Deep Learning Approach to Enhancing Therapy Chatbot Conversations.” Journal of Artificial Intelligence in Medicine, vol. 15, no. 2, pp. 89–104, 2023.

[12]. Wang, Yu, et al. “Exploring the Effectiveness of AI Chatbots in Providing Cognitive Behavioral Therapy for Anxiety Disorders.” International Journal of Medical Informatics, vol. 95, pp. 12–22, 2024.

[13]. Kim, Eun-Ju, et al. “Integration of Cognitive Behavioral Therapy Techniques into a Chatbot for Depression Support.” Computers in Human Behavior, vol. 78, pp. 132–141, 2025.

[14]. Lee, Seung-Hyun, et al. “Personalized Therapy Chatbot Using Reinforcement Learning and Natural Language Understanding.” IEEE Transactions on Neural Networks and Learning Systems, vol. 33, no. 5, pp. 1887–1900, 2026.

[15]. Zhang, Ming, et al. “Enhancing User Engagement with Therapy Chatbots through Adaptive Conversational Strategies.” International Journal of Human-Computer Studies, vol. 110, pp. 63–75, 2027.

[16]. Huang, Wei, et al. “User-Centered Design of a Therapy Chatbot for Stress Management: A Case Study.” Journal of Medical Internet Research, vol. 19, no. 8, e277, 2028.