SONORA: AN AI-DRIVEN JOURNALING APPLICATION FOR MOOD DETECTION AND PERSONALIZED MUSIC-BASED EMOTIONAL SUPPORT

Isabella Lai [1] , Jonathan Thamrun [2]

[1]The Winsor School, 103 Pilgrim Road, Boston, MA 02215
[2]University of California, Irvine, Irvine, CA 92697

ABSTRACT

Journaling is a proven method to improve mental well-being, but many people struggle to make it a consistent habit [1]. To address this, the Sonora application was developed to integrate journaling with personalized music playlists to enhance mental well-being. The app uses AI to analyze journal entries, detect mood, and generate a playlist that either matches or uplifts the user's emotional state. The app uses Flutter for the interface, Firebase for backend authentication, and OpenAI to power the mood analysis system [2]. Three core systems– journal entry processing, playlist generation, and user interface– work together seamlessly. A major challenge was interpreting complex human emotions accurately solely through words. In an experiment of 20 test journal entries, the system achieved 90% accuracy in mood detection. By combining music therapy with journaling, Sonora offers a unique, evidence-based way for users to reflect and improve mood [3].

KEYWORDS

Mental Wellbeing, Mood Detection, Music Therapy, AI Applications

1. INTRODUCTION

Journaling is one of those habits that spark the interest of many people looking to improve their well-being but rarely end up being a consistent part of one's daily routines. However, journaling is scientifically backed to have many positive benefits to general well-being and stress relieving. This motivates the development of an approach aimed at making journaling more appealing to a broader audience, including younger users. By integrating journaling with music, the system is designed to associate journaling behavior with positive reinforcement through music discovery, potentially supporting habit formation. The additional built in interactive streaks, mood logs, and more features will add to the appeal of the app.

Section 5 reviews three scholarly sources detailing existing emotional regulation applications. The first, Efficacy and Outcomes of a Music-Based Emotion Regulation Mobile App in Distressed Young People by Leanne Hides, provides personalized music for emotional support but lacks integration with the journaling [4]. The second, Mobile Apps That Promote Emotion Regulation, Positive Mental Health, and Well-being in the General Population by Mia Eisenstadt et al., surveys apps using static prompts or exercises which can feel generic or robotic for users. Third, Musicotherapy Mobile Applications: What Level of Evidence and Future Perspectives? Reviews music-based apps for anxiety and depression that generally provide soundscapes without analyzing

David C. Wyld et al. (Eds): ACSTY, MLSC, SVC, AIBD, ITCSS, ADCOM, NATP, SOFE – 2026
pp. 179-187, 2026. CS & IT - CSCP 2026 DOI: 10.5121/csit.2026.160414

user input. Sonora differs from these approaches by combining real-time journaling with AI-driven mood detection and personalized playlist generation [5]. This creates an interactive experience.

Unlike the other apps, Sonora can tailor feedback to the user's current emotions which offers a far more comprehensive and effective tool for emotional regulation and well-being.

Obtaining a tailored playlist of songs to improve mood within seconds of journaling will increase the appeal of this mundane but beneficial habit. Essentially, the user will do their journaling, select whether to match or uplift their mood, customize some song genres, then receive a personal playlist tailored to their mood. The app also has interactive features like streaks, mood logs, and more to add to the appeal. Currently, there aren't many online apps that make journaling as interactive as Sonora, much less integrate music with well-being. And compared to current solutions that exist, not only is music appealing to huge communities, but music in itself also has therapeutic effects, so everything is working together.

In this experiment, the accuracy of the application's AI system was evaluated, which analyzes journal entries and generates playlists based on the user's mood. A dataset of 20 journal entries (n = 20) representing a range of emotional states was constructed, both positive and negative, and compared the AI's detected mood with my expected results. The system correctly identified 18 out of the 20 entries, achieving an accurate rate of 90%. The main inaccuracies occurred in entries that had mixed emotions or vague wording, such as "I felt sick but got better later," which made mood detection more complex. These results suggest that while the AI performs well when emotions are relatively simple and clear, it struggles with nuanced or shifting emotional tones [6]. Overall, the experiment was successful, as it demonstrated the system's very strong baseline of accuracy, and it also provided insight into where further improvements in analysis are needed.

2. CHALLENGES

In order to build the project, a few challenges have been identified as follows.

2.1. Mood Detection Accuracy Concerns

Potential concerns include the ability of text-based analysis to capture complex emotional states with the argument that simply analyzing journal text may not always capture the full range of complex emotions. They may also wonder how consistent results are across different writing styles. To address this, the system was carefully designed to recognize a broad range of emotional cues and word patterns commonly used in journaling.

2.2. Playlist Recommendation Accuracy

One might also question how accurately the playlist recommendation will reflect the user's mood or taste. They might claim that the recommendations are generic or random. To address this, the system uses a hybrid approach to match the playlist to both the detected emotional tone and the user's customized genres. By combining the mood classification with personal preference, the system will ensure that the playlist suggestions feel tailored and not automated. The user also has the option to select whether to "match" or "uplift" their mood, further enhancing personal control.

2.3. Sustaining User Engagement

One might argue that journaling apps are all similar, and many will lose interest in the habit due to boredom. The app is not only designed with simplicity and elegance, but also has interactive

features like streaks, mood logs, and playlist results are implemented to keep the user engaged, with the core purpose to build the habit of journaling in order to spread its scientific benefits.

3. SOLUTION

When the program is first opened, the user is greeted with the splash screen, which briefly displays the app name and logo. During this gesture, the code is establishing a connection with the Firebase through the firebase_core integration. This makes sure that the user's authentication status, Firebase database access, and other data are properly configured before they proceed in the app [7]. From here, the user is directed to either the sign up or login page. A new user will be prompted to create an account, where then they are met with the homescreen. After the user has created an account for the first time, they will not see the login/signup screen every time they open the app, unless they change their password or run into other technical issues. Once on the homescreen, multiple options are available: starting a new journal entry, seeing all past entries, past playlists, and their mood tags with corresponding entries. The main feature is the journal. Once the user is finished writing, they press save, which prompts them to a selection screen. They are able to choose whether to match or uplift their mood and also select any genres of songs they want in their playlist. Here, the AI will analyze the user's entry, determine a mood from the list of pre-set moods, and generate a playlist based on the user's customizations.

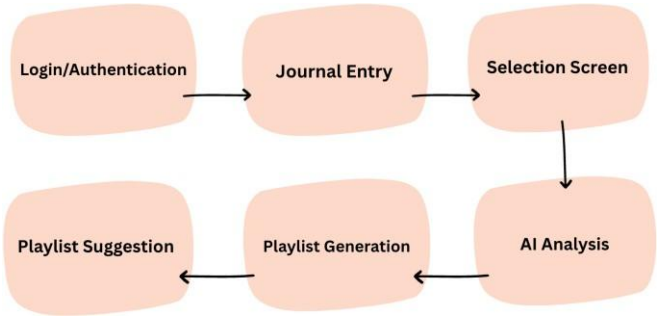

Figure 1. Overview of the solution

The playlist generation component analyzes the user's journal entry and pushes music recommendations based on the user's mood and customizations. It relies on the Firebase Firestore to store entries and a backend Cloud Function that runs the Neutral Language Processing. NLP identifies moods from a pre-set list and generates a playlist that either matches or uplifts the user's mood [8].

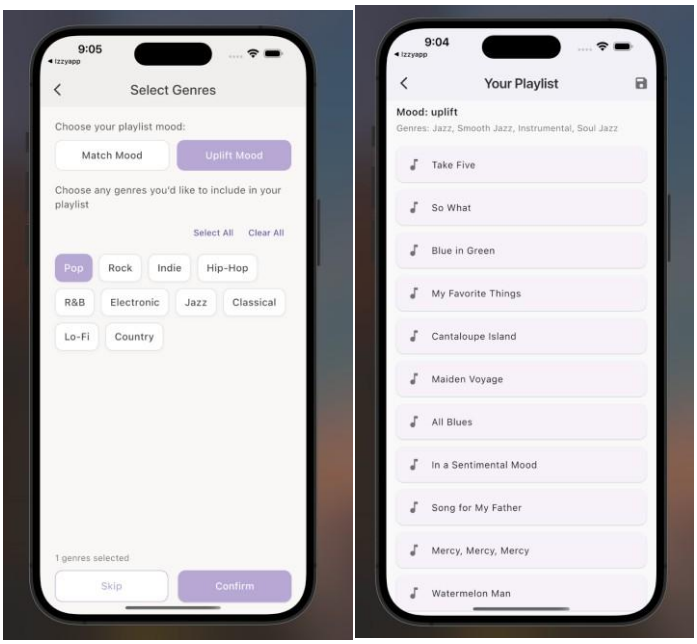

Figure 2. User interface of the Sonora application during journal interaction.

```
final resp = await http.post(
  Uri.parse(_endpoint),
  headers: {
    'Authorization': 'Bearer $idToken',
    'Content-Type': 'application/json',
  },
  body: jsonEncode({
    'journalEntryId': entryId,
    'journalText': text,
    'moodType': moodType,
    'selectedGenres': selectedGenres,
  }),
);

if (resp.statusCode != 200) {
  throw Exception('Server ${resp.statusCode}: ${resp.body}');
}

final data = jsonDecode(resp.body) as Map<String, dynamic>;
final playlist = (data['playlist'] ?? {}) as Map<String, dynamic>;
final songs = (playlist['songs'] as List?)
    ?.map<String>((s) => s['title']?.toString() ?? '')
    .toList();

if (!mounted) return;
Navigator.pushReplacement(
  context,
  MaterialPageRoute(
    builder: (_) => PlaylistSuggestionScreen(
      moodType: moodType ?? playlist['title']?.toString(),
      selectedGenres: (playlist['genres'] as List?)
          ?.map((e) => e.toString())
          .toList(),
      songs: songs,
    ),  // PlaylistSuggestionScreen
  ),  // MaterialPageRoute
```

Figure 3. Backend request workflow for AI-based playlist generation

This selected code is responsible for the process of generating a playlist from the journal entry. First, it sends a POST request to the Cloud Function backend using the http package. This request includes the user's journal ID, the content of the entry, and any customizations on the selection screen, all encoded in JSON [9]. Authorization is provided with the Firebase ID tokens to ensure that the user is authenticated. Once the request is sent, the code checks the server's response. If the response status is not 200, it will indicate an exception that the request has failed. If successful, it will parse the JSON response to extract the playlist data. Essentially, it will convert the list of song objects into a simple list of song titles. Finally, the code navigates to PlaylistSuggestionScreen, passing the generated playlist, detected mood, and selected genres. This chunk of code shows the interactions between the frontend, backend, and AI components.

The PlaylistSuggestion component displays the generated playlist that the AI creates based on the user's journal. The user can look through the result, and when they return to the home screen, the playlist is automatically saved in the All Playlists section.

```
final Playlist currentPlaylist = Playlist(
  title: "Generated Playlist",
  mood: mood,
  date: DateTime.now(),
  songs: songs ?? [],
); // Playlist

return Scaffold(
  backgroundColor: const Color(0xFFF9F9F9),
  appBar: AppBar(
    title: const Text(
      'Your Playlist',
      style: TextStyle(
        fontSize: 22,
        fontWeight: FontWeight.w600,
        color: Color(0xFF2E2E2E),
      ), // TextStyle
    ), // Text
    centerTitle: true,
    backgroundColor: Colors.white,
    elevation: 0.5,
    iconTheme: const IconThemeData(color: Color(0xFF2E2E2E)),
    actions: [
      if (songs != null && songs!.isNotEmpty)
        IconButton(
          icon: const Icon(Icons.save, color: Colors.black54),
          onPressed: () {
            if (onSave != null) onSave!(currentPlaylist);
            ScaffoldMessenger.of(context).showSnackBar(
              const SnackBar(content: Text('Playlist saved!')),
            );
          },
        ) // IconButton
```

Figure 4. Data structure and storage process for generated playlists

This code first creates a "Playlist" object called currentPlaylist. It stores the playlist's title, the mood of the playlist, date/time, and a list of songs. The songs ?? [] ensures that if there are no songs, it will create an empty list rather than causing an error. Next, the code checks if the "songs" list exists and is not empty. If this condition is true, it displays a save icon in the app bar. When the user taps this icon, the code calls the optional "onSave" function, which passes "currentPlaylist" to it. This allows the parent component to save the playlist in Firebase Firestore. Finally, the code shows a "SnackBar" with a message confirming that the playlist has been saved. This gives immediate feedback to the user.

The genre selection component allows the users to customize their playlist by choosing which music genres to include. It does not require any backend services or AI. It just handles the user inputs and updates the UI.

```
Expanded(
  child: SingleChildScrollView(
    child: Wrap(
      spacing: 10,
      runSpacing: 10,
      children: genres.map((genre) {
        final isSelected = selectedGenres.contains(genre);
        return AnimatedContainer(
          duration: const Duration(milliseconds: 200),
          curve: Curves.easeInOut,
          padding: const EdgeInsets.symmetric(horizontal: 14, vertical: 10),
          decoration: BoxDecoration(
            color: isSelected ? const Color(0xFFB6A6D4) : Colors.white,
            borderRadius: BorderRadius.circular(12),
            border: Border.all(
              color: isSelected ? const Color(0xFFB6A6D4) : const Color(0xFFE0E0E0),
              width: 1.2,
            ), // Border.all
            boxShadow: [
              BoxShadow(
                color: Colors.black.withOpacity(0.03),
                blurRadius: 5,
                offset: const Offset(0, 2),
              ), // BoxShadow
            ],
          ), // BoxDecoration
          child: GestureDetector(
            onTap: () => toggleGenre(genre),
            child: AnimatedScale(
              duration: const Duration(milliseconds: 200),
              scale: isSelected ? 1.05 : 1.0,
              child: Text(
                genre,
                style: TextStyle(
                  fontSize: 16,
                  fontWeight: FontWeight.w500,
                  color: isSelected ? Colors.white : const Color(0xFF444444),
```

Figure 5. Genre selection interface logic and interaction flow.

This part of the code builds the grid of selectable music genres in SelectionScreen. The Expanded widget makes sure it fills the whole page so there is no empty space. The SingleChildScrollView allows the user to scroll if the list exceeds the display of one screen. The Wrap arranges each genre as a small button with spacing between them. For each genre in "genres," it checks if selectedGenres contains it to determine selection. Each genre is displayed in AnimatedContainer that changes color and borders when toggled. Inside of the container, a GestureDetector wraps the genre Text, calling toggleGenre when it's tapped in order to add or remove the genre from selectedGenres. AnimatedScale slightly enlarges the text when selected for a visual cue [10]. This layout will allow the user to select multiple genres interactively and update the UI instantly.

4. EXPERIMENT

The accuracy of the AI system was evaluated to determine the reliability of mood-based playlist generation. It is important to measure accuracy because if the system misinterprets the user, the playlists will feel irrelevant to their entry or mood.

For this experiment, I created 20 journal entry inputs with a pretty clear and expected mood/theme. The inputs varied from happy and calm entries like "I took a walk in the park today," to sad or stressed ones like "I struggled on a test." After each entry there are also user customizations about genre and mood preferences for the playlist. It's important that the inputs are varied across a wide emotional range so that we know the AI is not biased towards one mood or theme.

	Input	Expected Output	Actual Output
1	The weather is pretty bad today	Negative	Negative
2	I had a great day!	Positive	Positive
3	I did bad on a test today	Negative	Negative
4	I had a usual day today	Positive/Neutral	Positive
5	I got into an argument with my parents	Negative	Negative
6	I felt nauseous and sick at school, but then I got better afterwords	Negative and Positive	Positive
7	I woke up late again and missed my bus. Today already feels ruined	Negative	Negative
8	Got an A on my Spanish test! All my studying paid off	Pos	Pos
9	I had a peaceful walk at sunset with my family. The view was beautiful	Pos	Pos
10	Taylor Swift dropped a new album and it was really good	Pos	Pos
11	I feel really anxious about my future and college applications	Neg	Neg
12	Our group project actually went well today even though I stayed up late working on it.	Pos	Pos
13	It's been raining for the past 2 days, I feel gloomy	Neg	Neg
14	I made banana bread with my sister and it turned out perfect	Pos	Pos
15	I went to the gym at 5am today even though it took a lot of willpower. I'm proud of myself	Pos	Pos
16	I feel very behind when everyone else around me seems like they are on top of their lives	Neg	Neg
17	I'm very grateful for my family	Pos	Pos
18	I tried meditating for the first time today and it felt good	Pos	Pos
19	I think I bombed a test today. I'm very anxious	Neg	Neg
20	I tried painting today, but I didn't like it and found it boring	Neg	Neg

Figure 6. Sample journal entry inputs used for evaluating mood detection accuracy

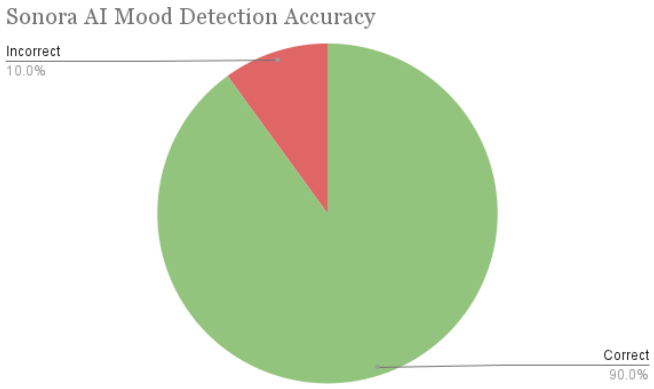

Figure 7. AI-predicted mood labels compared with expected mood categories for test journal entries

Out of the 20 inputs, the AI was able to correctly identify 18, with 2 that had more ambiguous generations due to many factors. This put the overall accuracy as 90%. When the entry had mixed emotions, or a change of emotions, it was harder for the AI to identify a single mood, making the output ambiguous. For instance, the entry "I felt nauseous and sick at school but felt better aftwards" indicates a change in mood, making the analysis more complicated. The other entries that ran into issues were shorter ones that lacked information. For instance, "I had a usual day today" may make

sense for the user, but the analysis won't be able to retrieve any emotional information from that entry alone. Despite this, the model performed consistently well with clearly positive or negative phrases, showing that it captures straightforward emotional language effectively. With more data and fine-tuning, the system could improve its accuracy on subtle or ambiguous emotions.

5. RELATED WORK

The research paper Efficacy and Outcomes of a Music-Based Emotion Regulation Mobile App in Distressed Young People: Randomized Controlled Trial by Leanne Hides outlines an app called Music eScape that is dedicated to teaching young people how to identify and manage emotions using music [11]. The mobile app offers personalized music tracks based on users' emotional states. While this app focuses on music-based emotion regulation, Sonora integrates journaling with mood detection and personalized music playlists. Our approach combines two therapeutic fields of journaling and music and integrates it into one minimal app. This offers a multifaceted approach to general well-being, offering more effective emotional regulation.

The study Mobile Apps That Promote Emotion Regulation, Positive Mental Health, and Well-being in the General Population by Mia Eisenstadt et al. reviews dozens of mental-health apps that support emotion regulation [12]. While these apps showed moderate success, they often relied on static prompts or generic exercises. In contrast, Sonora combines journaling and personalized music generation to create a dynamic and interactive feedback loop. Instead of just passively recording mood, users reflect through their entry and instantly receive a playlist that matches or uplifts their emotions. This personal touch makes my app more effective.

The paper Musicotherapy Mobile Applications: What Level of Evidence and Future Perspectives? published in Frontiers in Psychiatry explores mobile apps that use music for emotional healing, particularly towards anxiety and depression [13]. Most of these app's focus on providing soundscapes without considering real-time mood inputs from the user. Sonora is able to detect the user's mood through analysis of their entry before recommending a tailored playlist.

6. CONCLUSIONS

One of the main limitations of this project is the depth of the mood detection system. Since AI currently can only determine emotions based solely on text input, it can easily misinterpret tone, sarcasm, or more nuanced/complex emotions. Additionally, the playlist generation system depends on pre-defined song data rather than a real-time music API, which limits some personalization and song choice diversity [14]. Future work includes improving the emotion analysis model through training on larger and more diverse datasets. Future system extensions may include integration with external music APIs (e.g., Spotify) to enhance playlist diversity and real-time playback functionality [15]. In the future, Sonora could be expanded into a social platform where users can share their playlists, journal excerpts, or stats for connections and encouragement.

REFERENCES

[1] Hayman, Brenda, Lesley Wilkes, and Debra Jackson. "Journaling: Identification of challenges and reflection on strategies." Nurse researcher 19.3 (2012).

[2] Białkowski, Damian, and Jakub Smołka. "Evaluation of Flutter framework time efficiency in context of user interface tasks." Journal of Computer Sciences Institute 25 (2022): 309-314.

[3] Hillecke, Thomas, Anne Nickel, and Hans Volker Bolay. "Scientific perspectives on music therapy." Annals of the New York Academy of Sciences 1060.1 (2005): 271-282.
[4] Hides, Leanne, et al. "Efficacy and outcomes of a music-based emotion regulation mobile app in distressed young people: randomized controlled trial." JMIR mHealth and uHealth 7.1 (2019): e11482.
[5] Fatima, Eram, Namrata Dhanda, and Taskeen Zaidi. "AI-Driven Detection of Stress, Anxiety, and Depression: Techniques, Challenges, and Future Perspectives." 2025 3rd International Conference on Disruptive Technologies (ICDT). IEEE, 2025.
[6] Marechal, Catherine, et al. "Survey on AI-Based Multimodal Methods for Emotion Detection." High- performance modelling and simulation for big data applications 11400 (2019): 307-324.
[7] Moroney, Laurence. "An introduction to firebase." The Definitive Guide to Firebase: Build Android Apps on Google's Mobile Platform. Berkeley, CA: Apress, 2017. 1-24.
[8] Mishra, Asha Rani, et al. "Unveiling Emotions: NLP-Based Mood Classification and Well-Being Tracking for Enhanced Mental Health Awareness." Mathematical Modelling of Engineering Problems 12.2 (2025).
[9] Bourhis, Pierre, et al. "JSON: data model, query languages and schema specification." Proceedings of the 36th ACM SIGMOD-SIGACT-SIGAI symposium on principles of database systems. 2017.
[10] Jacobs, Robert A. "What determines visual cue reliability?." Trends in cognitive sciences 6.8 (2002): 345- 350.
[11] Hides, Leanne, et al. "Efficacy and outcomes of a music-based emotion regulation mobile app in distressed young people: randomized controlled trial." JMIR mHealth and uHealth 7.1 (2019): e11482.
[12] Eisenstadt, Mia, et al. "Mobile apps that promote emotion regulation, positive mental health, and well-being in the general population: systematic review and meta-analysis." JMIR mental health 8.11 (2021): e31170.
[13] Leschallier De Lisle, Gaëtan, et al. "Musicotherapy mobile applications: what level of evidence and potential role in psychiatric care? A systematic review." Frontiers in Psychiatry 15 (2024): 1366575.
[14] Tsuchiya, Takahiko, Jason Freeman, and Lee W. Lerner. "Data-to-music API: Real-time data-agnostic sonification with musical structure models." Georgia Institute of Technology, 2015.
[15] Anderson, Ian, et al. "“Just the way you are”: Linking music listening on Spotify and personality." Social Psychological and Personality Science 12.4 (2021): 561-572.