

EGO-PSY: A PERSONALIZED AND ACCESSIBLE MOBILE PLATFORM FOR HOLISTIC MENTAL HEALTH SUPPORT

Ziang Wang ¹, YuCao ²

¹ Stuart country day school of sacred heart, 1200 Stuart Rd,
Princeton, NJ 08540

² California State Polytechnic University, Pomona, CA, 91768

ABSTRACT

This paper presents Ego-psy, a mobile application designed to improve accessibility and personalization in mental health support [1]. The app integrates several features, including mood logging with AI-generated feedback, medication tracking, psychology-related courses, and a searchable database of professionals. Using Firebase Authentication and Firestore, the app ensures secure data handling while maintaining scalability [2]. Three major systems—course creation, authentication, and AI feedback—are analyzed in detail. An experiment was conducted using a survey of five participants, which revealed strong usability and satisfaction scores, particularly for mood logging and design, while also identifying the improvement areas such as motivation and medical features. Comparisons with existing research demonstrated how Ego-psy builds upon proven strengths of conversational agents while addressing their shortcomings by offering more holistic and ethical support. Limitations and future improvements are also discussed. Overall, Ego-psy provides a unique, multi-faceted approach to digital mental health, emphasizing accessibility, engagement, and safety [3].

KEYWORDS

Mental Health, Personalization, Accessibility, Artificial Intelligence

1. INTRODUCTION

Mental health issues have become increasingly prevalent worldwide, with anxiety and depression ranking among the most common disorders. According to the World Health Organization, approximately one in eight people globally live with a mental health condition, yet access to professional support remains limited due to stigma, cost, and shortage of trained providers. Many individuals lack the resources or confidence to seek therapy, which results in untreated symptoms, reduced quality of life, and significant social and economic burdens. Young adults and teenagers are particularly affected, as they often face academic stress, social pressure, and uncertainty about the future while also being more likely to engage with technology-based solutions [4]. Furthermore, while mental health apps exist, many focus narrowly on meditation or tracking symptoms, without offering personalized, AI-driven feedback that encourages reflection and growth. The problem, therefore, lies in the gap between the rising demand for accessible mental health support and the limited supply of tailored, interactive tools that foster daily engagement. Addressing this issue is critical, as early and consistent self-care can prevent conditions from worsening, empower users to better understand their emotions, and complement professional care when available.

In the methodology section, we compared Ego-psy to three research projects in the mental health AI space [5]. The first, by Li et al. (2023), reviewed conversational agents for mental health and found they reduce depression and distress, though overall well-being gains were modest. Ego-psy improves on this by adding medication tracking, educational courses, and professional support features, broadening its impact beyond symptom reduction. The second, EMMA by Ghandeharioun et al. (2021), is an emotion-aware chatbot that delivers micro-interventions based on detected moods. While effective, EMMA lacks journaling, feedback, and health management features that Ego-psy provides. The third, by Dohnány et al. (2025), raised concerns about feedback loops where users form unhealthy dependencies on chatbots. Ego-psy addresses this risk by framing AI as supportive but not a replacement for care, and by including grounding tools such as doctor connections. These comparisons highlight Ego-psy's holistic, safer, and more versatile design.

Ego-psy was developed as a mobile application designed to address the challenges of accessibility and personalization in mental health support. The app integrates mood logging, AI-generated feedback, medication tracking, course recommendations, and professional resources into a single platform. The central idea is to provide users with a safe, private space to reflect on their daily emotions while receiving supportive feedback from a large language model. Unlike static journaling apps, Ego-psy uses generative AI to respond to user entries, making the experience feel conversational and personalized. Additionally, the app includes a medication tracker that allows users to record dosages and view history, helping them stay organized in their treatment routines. The courses feature psychology-related learning material, empowering users with knowledge that can complement therapy. To ensure security and reliability, Firebase Authentication and Firestore handle user data and storage [6]. Ego-psy's uniqueness lies in combining self-reflection, education, and AI-driven support in one system, rather than focusing on only one aspect of mental health. This holistic approach positions the app as a more effective tool compared to existing solutions. By lowering barriers to self-care, Ego-psy empowers individuals to actively monitor their well-being, fostering consistency and early intervention.

For the experiment, we conducted a user survey to evaluate how well Ego-psy fits with a general audience. Five participants tested the app's main features, including mood logging with AI feedback, medication tracking, course browsing, and doctor search. They then completed a 10-question survey, rating their experience on a 1–5 scale. The survey results showed strong performance in navigation, mood logging, and design, with averages above 4.5, while AI feedback also scored positively at 4.4. However, features such as medication tracking and doctor search received lower averages around 3.8–3.9, highlighting areas for improvement. Motivation to log emotions regularly was the lowest-rated category at 3.6, suggesting the need for engagement strategies. Overall satisfaction was rated 4.4, and most participants said they would recommend the app. These findings confirm that Ego-psy provides meaningful support but should improve long-term engagement and medical tools for maximum impact.

2. CHALLENGES

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

2.1. Course Creation and Storage

A potential concern with the course creation system is whether user-generated content will be reliable or helpful. To address this, only authorized psychologists are allowed to create courses, and authorization requests are carefully verified through the profile system. Another skepticism is whether storing large amounts of course data in Firebase might create performance issues, but

Firestore is optimized for scalable, real-time applications and the data is encoded efficiently to ensure smooth performance. Some may also question why users would choose these courses over freely available online resources. The difference lies in the integration: Ego-psy courses are designed specifically for mental health support and are embedded within a broader ecosystem of tools, making the learning experience more personalized and contextually relevant.

2.2. Authentication

Concerns about the authentication system typically focus on security. Some might wonder whether Firebase Authentication is strong enough to protect sensitive user information. Since Firebase is an industry-standard service that offers encryption and compliance with security protocols, users can trust that their accounts are safe. Another worry is account recovery if users forget their passwords or lose access. This is mitigated by Firebase's built-in password reset and recovery features. Others may raise the possibility of unauthorized access or hacking. All user data is encrypted and tied to unique credentials, and the app can be expanded with multi-factor authentication in future updates for even greater protection.

2.3. AI Feedback

The AI feedback system may raise doubts about accuracy and reliability. Critics might question whether AI-generated feedback can truly provide meaningful emotional support. To address this, Ego-psy's generative model is guided by carefully constructed prompts designed to encourage positive, empathetic responses rather than clinical advice. Another concern is that AI might misinterpret ambiguous or sarcastic statements. While this limitation exists, the app frames the AI feedback as supportive guidance rather than definitive analysis, which reduces the risk of harmful misclassification. Finally, skeptics could argue that AI feedback may discourage users from seeking professional help. To counter this, Ego-psy positions its AI as a complement to — not a replacement for — psychologists, with clear pathways for users to connect with professionals through the doctor search feature.

3. SOLUTION

When entering the app, the user will be greeted with the sign up / login screen. After logging in or creating an account, the user will then be redirected to the home page.

On the home page, the user can log their day, track their medication, read recommended courses, and navigate to other apps.

When logging their day, the user will be prompted to enter a log entry, along with their mood for the day. After that, a large language model would generate feedback for the user's log for the user to view. Users can view their logs inside the history page.

To track their medication, the user can search up the medicine they are taking from a list of medicines related to mental illness, and enter a dosage and extra notes if applicable. In the history page, users can view their past medication history, and ask AI for a weekly feedback.

The user can learn courses related to psychology from recommended courses on the home page, or from the course page directly. To create courses, the user has to become an authorized user which can be done by going to profile page and requesting to be an authorized psychologist [7].

On the doctor page, users can search for psychological experts's assistance. They can view the expertise and contact info of the doctors by clicking into their profiles.

In the profile page, the user can view and update their profile information, request to be authorized, log out, and delete their account.

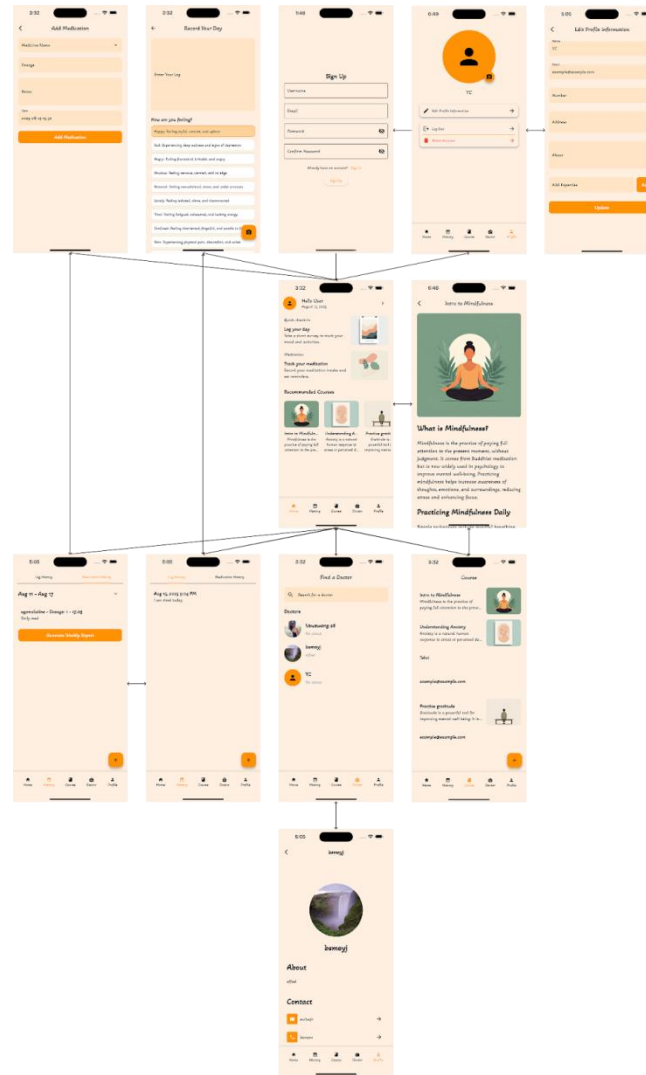


Figure 1. Overview of the solution

Ego-psy allows authorized users to create their own courses, each course can contain different sections like image, subsection title, and subsection content. To store this course object inside firebase efficiently, the data can be encoded before storing. This code section below shows the encoding and uploading process of a newly created course.



```

1
2 Future<bool> addCourse(
3     List<CourseSection> sections,
4     String title,
5     BuildContext context,
6 ) async {
7     try {
8         final courseRef = FirebaseFirestore.instance.collection('courses').doc();
9         Map<String, String> sectionContent = {};
10        for (int i = 0; i < sections.length; i++) {
11            if (sections[i].type == SectionType.title) {
12                sectionContent[i.toString() + "_" + sections[i].type.name] =
13                    sections[i].boldText;
14            }
15            if (sections[i].type == SectionType.description) {
16                sectionContent[i.toString() + "_" + sections[i].type.name] =
17                    sections[i].description;
18            }
19            if (sections[i].type == SectionType.image) {
20                final imageUrl = await uploadCourseImage(
21                    sections[i].image,
22                    courseRef.id,
23                    i.toString(),
24                    context,
25                );
26                sectionContent[i.toString() + "_" + sections[i].type.name] = imageUrl;
27            }
28        }
29        await courseRef.set({
30            'title': title,
31            'createdAt': DateTime.now(),
32            'from': FirebaseAuth.instance.currentUser!.uid,
33            'sections': sectionContent,
34        });
35        ScaffoldMessenger.of(context).showSnackBar(
36            const SnackBar(content: Text("Course added successfully")),
37        );
38        return true;
39    } catch (e) {
40        ScaffoldMessenger.of(context).showSnackBar(
41            SnackBar(content: Text("Failed to add course $e")),
42        );
43        return false;
44    }
45 }

```

Figure 2. Screenshot of code 1

This `addCourse` is designed to add a new course to Firestore [8]. It starts by creating a new document reference in the 'courses' collection to obtain the id which will be used to upload course images. It then initializes an empty map to store the processed section content.

The function then loops through each course section, handling different section types appropriately: for title sections, it extracts the bold text; for description sections, it gets the description text; and for image sections, it calls an upload function to store the image and retrieves the resulting URL. Each section is stored in the map with a key format of "index_type" (like "0_title" or "1_description"). This is for reconstructing the course with the correct order when obtaining the data from firebase.

Once all sections are processed, the function creates the course document in Firestore containing the title, creation timestamp, the current user's ID, and all the processed sections. If everything succeeds, it shows a success message to the user and returns true. However, if any errors occur during the process (like upload failures or Firestore errors), the function catches the exception, displays an error message with details, and returns false. This approach ensures robust error handling while providing clear feedback to users about the operation's success or failure.

Ego-psy uses Firebase Authentication to manage user login and authentication. This system is a critical component of the app because it ensures that all user data remains secure and private [9]. The current code section includes all the major Firebase Authentication functions that the app relies on including sign up, log in, sign out, and delete account.

```

1 import 'package:firebase_auth/firebase_auth.dart';
2 import 'package:flutter/material.dart';
3
4 Future<UserCredential?> signUpWithEmailAndPassword(
5   String emailAddress, String password, BuildContext context) async {
6   try {
7     return await FirebaseAuth.instance.createUserWithEmailAndPassword(
8       email: emailAddress,
9       password: password,
10     );
11   } on FirebaseAuthException catch (e) {
12     if (context.mounted) {
13       if (e.code == 'weak-password') {
14         ScaffoldMessenger.of(context).showSnackBar(const SnackBar(
15           content: Text('The password provided is too weak.')));
16       } else if (e.code == 'email-already-in-use') {
17         ScaffoldMessenger.of(context).showSnackBar(const SnackBar(
18           content: Text('The account already exists for that email.')));
19       } else {
20         ScaffoldMessenger.of(context)
21           .showSnackBar(SnackBar(content: Text(e.code)));
22       }
23     }
24   } catch (e) {
25     if (context.mounted) {
26       ScaffoldMessenger.of(context)
27         .showSnackBar(const SnackBar(content: Text('Error signing up')));
28     }
29   }
30   return null;
31 }
32
33 Future<UserCredential?> loginWithEmailAndPassword(
34   String emailAddress, String password, BuildContext context) async {
35   try {
36     return await FirebaseAuth.instance
37       .signInWithEmailAndPassword(email: emailAddress, password: password);
38   } on FirebaseAuthException catch (e) {
39     if (context.mounted) {
40       if (e.code == 'user-not-found') {
41         ScaffoldMessenger.of(context).showSnackBar(
42           const SnackBar(content: Text('No user found for that email.')));
43       } else if (e.code == 'wrong-password') {
44         ScaffoldMessenger.of(context).showSnackBar(const SnackBar(
45           content: Text('Wrong password provided for that user.')));
46       } else {
47         ScaffoldMessenger.of(context)
48           .showSnackBar(SnackBar(content: Text(e.code)));
49       }
50     }
51   } catch (e) {
52     if (context.mounted) {
53       ScaffoldMessenger.of(context)
54         .showSnackBar(const SnackBar(content: Text('Error logging in')));
55     }
56   }
57   return null;
58 }
59
60 Future<void> signOut() async {
61   await FirebaseAuth.instance.signOut();
62 }
63
64 Future<void> deleteAccount() async {
65   await FirebaseAuth.instance.currentUser!.delete();
66 }
67

```

Figure 3. Screenshot of code 2

This code contains the four main authentication functions for Firebase: sign up, login, sign out, and delete account. The first function `signUpWithEmailAndPassword` handles user registration by calling Firebase's `createUserWithEmailAndPassword` method [14]. It includes comprehensive error handling for common Firebase authentication exceptions: `weak-password` and `email-already-in-use`. It also displays appropriate user-friendly error messages through snackbars if there is an error.

The second function `loginWithEmailAndPassword` manages user login by calling Firebase's `signInWithEmailAndPassword` method. Similar to the signup function, it handles specific Firebase authentication errors like `user-not-found` and `wrong-password`. Againsimilar to the

signup function, it also displays appropriate user-friendly error messages through snackbars if there is an error.

Both functions include a safety check using `context.mounted` to ensure the context is still valid before showing snackbar messages, which prevents crashes if the widget is disposed before the async operation completes.

The third function `signOut` is a simple wrapper that calls Firebase's `signOut` method.

The fourth function `deleteAccount`, similar to the signup and login functions, is also a wrapper that calls Firebase's `delete` method on the currently authenticated user.

When a user creates an emotion log, they will provide a text entry, select an emotion, and may upload an optional image. After the log submission, AI will generate personalized feedback based on the user's inputs. The feedback along with the log details will be stored in Firebase Firestore so users can revisit their entries and track their emotional history over time. The functions below show how the AI feedback is generated and saved.



```

1 Future<void> _generateFeedback() async {
2   try {
3     final model = GenerativeModel(
4       model: 'model',
5       apiKey: _apiKey,
6     );
7
8     String promptText = ''
9     prompt
10    '';
11
12    final response = widget.logImage != null
13      ? await _generateImageResponse(model, promptText)
14      : await _generateTextResponse(model, promptText);
15
16    Provider.of<AddLogProvider>(context, listen: false)
17      .addAIResponseText(response);
18
19    setState(() {
20      _feedback = response;
21      _isLoading = false;
22    });
23  } catch (e) {
24    setState(() {
25      _feedback = 'Error generating feedback: $e';
26      _isLoading = false;
27    });
28  }
29 }
30
31 Future<String> _generateTextResponse(
32   GenerativeModel model, String promptText) async {
33   final response = await model.generateContent([
34     Content.text(promptText),
35   ]);
36   return response.text ?? 'Unable to generate feedback at this time.';
37 }
38
39 Future<String> _generateImageResponse(
40   GenerativeModel model, String promptText) async {
41   final bytes = await widget.logImage!.readAsBytes();
42
43   final response = await model.generateContent([
44     Content.text(promptText),
45     Content.multi([
46       TextPart(promptText),
47       DataPart('image/jpeg', bytes),
48     ]),
49   ]);
50
51   return response.text ?? 'Unable to generate feedback at this time.';
52 }

```

Figure 4. Screenshot of code 3

The `generateFeedback` function is used to create AI-generated feedback for a user's log entry. It begins by instantiating a `GenerativeModel` with the configured model name and API key for later use. Then prepares a prompt string to guide the model's response [10].

Depending on whether an image is attached to the log, it chooses between two specialized helpers. If there is no image, it calls `_generateTextResponse`, which sends a simple text prompt to the model and returns the resulting text or a safe fallback if the model does not produce any. If an image is present, it calls `_generateImageResponse`, which first reads the image bytes and then submits a multipart request that includes both the textual prompt and the image data so the model can reason over visual context.

Once a response is obtained, the method stores the text in application state via `AddLogProvider.addAIResponseText` and triggers rebuild by calling `setState`, setting the feedback string and clearing the loading flag.

The entire process is wrapped in a `try/catch` block so that any exception will trigger an error message to be shown in the UI and will also end the loading state.

4. EXPERIMENT

This experiment aims to test how well Ego-psy fits with a general audience by collecting feedback from users through a structured survey. By asking targeted questions about usability, effectiveness, and satisfaction, we can understand whether the app meets user expectations and supports their mental health needs.

We designed a survey consisting of 10 questions, each answered on a 1–5 Likert scale (1 = strongly disagree, 5 = strongly agree). The questions measure usability, clarity, usefulness, and overall satisfaction with the app. To conduct the experiment, we invited 5 participants to use Ego-psy for a short trial period, during which they explored its core functions such as mood logging, AI feedback, medication tracking, and course browsing. Afterward, participants completed the survey. The questions were chosen to evaluate whether the app was intuitive, engaging, and effective in supporting emotional well-being.

#	Question	Scale (1–5)
1	The app was easy to navigate.	1–5
2	The mood logging feature was useful.	1–5
3	The AI feedback felt relevant and supportive.	1–5
4	Tracking medication was simple and helpful.	1–5
5	The course recommendations were interesting.	1–5
6	The design and layout of the app were appealing.	1–5
7	I felt motivated to log my emotions regularly.	1–5
8	The doctor search feature was informative.	1–5
9	I would recommend this app to a friend.	1–5
10	Overall, I am satisfied with the app.	1–5

Figure 5. Question and scale

Question	Avg. Score	Notes
1. The app was easy to navigate.	4.6	Users praised the clean layout.
2. The mood logging feature was useful.	4.8	Most found it simple and meaningful.
3. The AI feedback felt relevant and supportive.	4.4	Some said feedback was "encouraging."
4. Tracking medication was simple and helpful.	3.8	A few wanted reminders/alerts.
5. The course recommendations were interesting.	4.0	Content seen as helpful but limited.
6. The design and layout of the app were appealing.	4.6	"Modern" and "calming" design praised.
7. I felt motivated to log my emotions regularly.	3.6	Mixed — some unsure about long-term use.
8. The doctor search feature was informative.	3.9	Good info, but not detailed enough.
9. I would recommend this app to a friend.	4.2	Positive overall sentiment.
10. Overall, I am satisfied with the app.	4.4	Majority gave 4 or 5 stars.

Figure 6. Question and notes

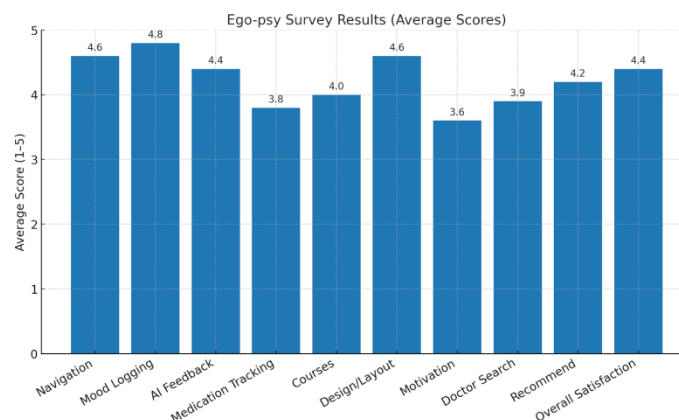


Figure 7. Figure of experiment

The survey results showed that participants generally had a positive experience using Ego-psy. Navigation, design, and mood logging received the strongest ratings (average scores above 4.5), demonstrating that the app is intuitive and visually appealing. The AI feedback feature also scored well (4.4), with participants noting that responses felt encouraging and personal. However, medication tracking and doctor search scored lower (3.8 and 3.9), indicating areas for improvement. Several participants suggested adding reminders for medication and expanding doctor profiles with more details. Motivation for long-term mood logging received the lowest rating (3.6), which suggests the app may need gamification or habit-forming features to encourage regular use. Overall satisfaction was high (4.4), and most participants said they would recommend the app to others. These findings indicate that while Ego-psy is effective in delivering supportive features, enhancing engagement and medical tools could make it more impactful.

5. RELATED WORK

In the systematic review titled “Systematic review and meta-analysis of AI-based conversational agents for promoting mental health and well-being,” authors Han Li, Renwen Zhang, Yi-Chieh Lee, Robert E. Kraut, David C. Mohr, and colleagues analyze how AI-driven conversational agents perform in mental health interventions [11]. Their project aggregates data from randomized controlled trials and experimental studies, finding that such agents—especially those employing generative, multimodal AI via mobile platforms—effectively reduce symptoms of depression (Hedge’s $g \approx 0.64$) and psychological distress (Hedge’s $g \approx 0.70$), although improvement in overall well-being was modest (Hedge’s $g \approx 0.32$). In contrast, Ego-psy functions as an integrated mobile app combining mood logging, AI feedback, medication tracking, educational courses, and doctor search into one ecosystem. Unlike the generalized conversational agents evaluated in the review, Ego-psy’s system blends self-reflection tools with personalized generative feedback and practical health features. This holistic integration, especially the inclusion of medication tracking and curated educational content, is unique compared to the research-reviewed CA systems. I believe Ego-psy offers a more engaging and context-sensitive user experience, not only addressing symptom reduction through conversational AI but also encouraging daily self-management, education, and connection to care, fostering broader, more sustained mental health support than the standalone agents evaluated in the meta-analysis.

The study “EMMA: An Emotion-Aware Well-being Chatbot” by Asma Ghandeharioun, Daniel McDuff, Mary Czerwinski, and Kael Rowan (2021) presents a thoughtfully designed system that delivers just-in-time, emotionally tailored micro-interventions by detecting users’ moods via smartphone sensors over a two-week trial with 39 participants [12]. EMMA excels in seamlessly offering support rooted in real-time affective data and was found to be likable and appropriate in its emotional tone. However, EMMA does not allow users to express their own emotional states through journaling, nor does it offer AI-generated personalized feedback, educational resources, medication tracking, or pathways to care. In contrast, Ego-psy actively engages users through self-reflection: mood logging triggers conversational AI feedback, while integrating course content, medication management, and doctor search features round out its holistic design. As such, Ego-psy not only responds to sensed emotions but also empowers users to articulate, learn, and manage their mental health in a more comprehensive way. This broader functionality may strengthen user engagement, promote self-management skills, and facilitate progression toward professional care, advantages that position Ego-psy as a more multi-faceted tool compared to the more narrowly focused EMMA system.

An insightful recent study titled “Technological folie à deux: Feedback Loops Between AI Chatbots and Mental Illness” by Dohnány et al. (2025) examines the growing psychological risks associated with emotionally vulnerable individuals forming intense bonds with AI chatbots, highlighting cases of delusional thinking, reinforced harmful beliefs, and emotionally destabilizing interactions [13]. Ego-psy consciously addresses these issues by combining AI-generated emotional support with structured self-reflection, education, and clear pathways to professional care. Unlike the systems described in the Dohnány et al. study, which can inadvertently encourage harmful psychological feedback loops, Ego-psy encourages grounding through active mood logging, curated courses, and medication tracking that foster self-awareness and behavioral reinforcement rather than emotional dependency. Additionally, the app positions AI feedback as a supportive adjunct—not a standalone therapy—encouraging users to seek help when needed via the doctor search feature. This design mitigates risks related to “chatbot psychosis” or over-reliance on AI, offering a more balanced, safe, and comprehensive platform. By embedding reflective practices, educational tools, and links to real-world help in addition to AI interaction, Ego-psy pioneers a more ethically grounded and psychologically resilient approach to digital mental health support.

6. CONCLUSIONS

While Ego-psy demonstrates strong potential as a supportive mental health tool, there are several limitations to acknowledge. First, the AI feedback system, though generally accurate, sometimes struggles with nuanced emotional expressions such as sarcasm, indirect negativity, or cultural variations in language [15]. This could lead to occasional misclassifications or responses that feel less personalized. Second, the medication tracking feature is currently limited to manual input without reminders or integration with pharmacy systems, which may reduce its usefulness for users managing complex treatment schedules. Third, while the course feature provides valuable educational content, its scope is still narrow and depends on the availability of authorized contributors, which may limit diversity and depth of material. Additionally, because the app relies heavily on self-reported data, user engagement is critical, and motivation to log moods regularly may decline over time.

Looking ahead, several improvements could strengthen the platform. Expanding the training data for the AI model would help it better capture subtle emotions and contextual meaning. Adding push notifications or calendar integration for medication tracking would increase adherence. Partnering with more licensed professionals to expand the course library would diversify the educational offerings. Finally, gamification features such as progress badges or streaks could improve user retention. Together, these enhancements would create a more robust, engaging, and effective tool for long-term mental health support.

REFERENCES

- [1] Repper, Julie, and Tim Carter. "A review of the literature on peer support in mental health services." *Journal of mental health* 20.4 (2011): 392-411.
- [2] Bačaniin Džakula, Nebojša. "Agile multi-user android application development with firebase: Authentication, authorization, and profile management." *Sinteza 2024-International Scientific Conference on Information Technology, Computer Science, and Data Science*. Singidunum University, 2024.
- [3] Lattie, Emily G., Colleen Stiles-Shields, and Andrea K. Graham. "An overview of and recommendations for more accessible digital mental health services." *Nature Reviews Psychology* 1.2 (2022): 87-100.
- [4] Reddy, K. Jayasankara, Karishma Rajan Menon, and Anjana Thattil. "Academic stress and its sources among university students." *Biomedical and pharmacology journal* 11.1 (2018): 531-537.
- [5] Spiegel, Brennan MR, et al. "Feasibility of combining spatial computing and AI for mental health support in anxiety and depression." *npj Digital Medicine* 7.1 (2024): 22.
- [6] Moroney, Laurence. "Using authentication in firebase." *The Definitive Guide to Firebase: Build Android Apps on Google's Mobile Platform*. Berkeley, CA: Apress, 2017. 25-50.
- [7] Sammons, M. T., R. F. Levant, and R. U. Paige. "Prescriptive authority for psychologists." Washington, DC: American Psychological Association (2003).
- [8] Kesavan, Ram, et al. "Firestore: The nosql serverless database for the application developer." *2023 IEEE 39th International Conference on Data Engineering (ICDE)*. IEEE, 2023.
- [9] Moroney, Laurence. "Using authentication in firebase." *The Definitive Guide to Firebase: Build Android Apps on Google's Mobile Platform*. Berkeley, CA: Apress, 2017. 25-50.
- [10] Lu, Hongqian Karen. "Keeping your API keys in a safe." *2014 IEEE 7th International Conference on Cloud Computing*. IEEE, 2014.
- [11] Li, Han, et al. "Systematic review and meta-analysis of AI-based conversational agents for promoting mental health and well-being." *NPJ Digital Medicine* 6.1 (2023): 236.
- [12] Ghandeharioun, Asma, et al. "Emma: An emotion-aware wellbeing chatbot." *2019 8th international conference on affective computing and intelligent interaction (ACII)*. IEEE, 2019.
- [13] Dohnány, Sebastian, et al. "Technological folie\`a deux: Feedback Loops Between AI Chatbots and Mental Illness." *arXiv preprint arXiv:2507.19218* (2025).

- [14] Porter, Chris, M. Angela Sasse, and Emmanuel Letier. "Designing acceptable user registration processes for e-services." BISL, 2012.
- [15] Xu, Wenzhong, et al. "Artificial intelligence in constructing personalized and accurate feedback systems for students." *International Journal of Modeling, Simulation, and Scientific Computing* 14.01 (2023): 2341001.