

PET'S MIND: AN AI-POWERED MOBILE APPLICATION FOR PET CARE, HEALTH TRACKING, AND COMMUNITY SUPPORT

Le Chen¹, Yu Cao²

¹ Yorba Linda High School, 19900 Bastanchury Rd, Yorba Linda, CA 92886

² California State Polytechnic University, Pomona, CA, 91768

ABSTRACT

This research paper explores the development of Pet's Mind, a mobile application designed to improve pet care through real-time AI assistance, health tracking, and community interaction [1]. The problem addressed is the lack of affordable, comprehensive, and accessible tools for new pet owners, many of whom struggle to provide proper care due to limited knowledge or resources. Our methodology involved designing three core systems: the AI Nutrition Expert, the Health Tracking system, and the Community Forum. These systems were implemented using Flutter and Firebase, ensuring accessibility across platforms [2]. To evaluate effectiveness, we conducted a user survey that tested navigation, usefulness, and satisfaction. The results showed high averages across most categories, particularly in usability and AI responses, with some room for improvement in design aesthetics and trust in AI accuracy. Overall, Pet's Mind offers a free, ad-free, and supportive platform that enables pet owners to make informed decisions and build healthier lives for their pets.

KEYWORDS

Pet Care, Mobile Application, AI Assistance, Health Tracking

1. INTRODUCTION

I am someone who has and loves raising pets. When people first get a pet, they might not have enough experience to keep the pet well and healthy, and this can be prevented if they can learn from others' experience. According to the NIH, 20% of the pet owners did no research before purchasing their animal, and 54% of owners had looked for advice or information before purchasing their dog. This shows the importance of pet knowledge among pet owners which inspired me to create a pet app to learn from other pet owners to ensure their pet's safety and well being. There are many pet related apps on the market, yet most of them are not free and require payment to access the app or premium features. However, not everyone who has a pet has the financial condition to pay for it, as a result, I want to create an easily accessible and ad-free pet app for the pet lovers. I think this is an important issue because according to the NIH, pet owners see their pets as their friend or part of their family, and they need to be taken care of well. We need to do our best to create an environment such that people's pets can live happily and healthily.

In the methodology section, we analyzed three related methodologies in pet care applications. The first was Diana's mobile app, which focused heavily on tracking detailed actions such as showering, vaccination, and deworming. While comprehensive, it lacked a community forum and real-time AI support. The second was Vanshri's pet care system, which combined training resources and doctor consultations but did not allow detailed pet tracking or instant nutritional feedback [3]. The third was Dr. More's web application, which used a different framework and emphasized web accessibility, but did not provide mobile convenience or a strong community aspect. Compared to these approaches, our app, Pet's Mind, stands out by combining essential tracking tools, real-time AI responses, and a community forum into one accessible mobile application [4]. By focusing on affordability, ease of use, and interactive features, our project improves upon these methodologies and provides a more complete solution for pet owners.

I created Pet's Mind because pets can often get sick if not well taken care of regularly. Speaking from experience, new pet owners often have no idea how to resolve issues that come up in raising their pet, which can cause critical health issues to their pets. Pet's Mind is an application that focuses on pet caring [5]. The app allows pet owners to create and track their pets by various metrics like weight, age, food intake, and activeness, which allows pet owners to develop a caring plan based on their pets' status to keep their pets healthy [6]. The app also includes an AI Nutrition Expert which will answer user's questions 24/7 in real time. Moreover, pet owners can share their experience with others on the community forum page. Other similar apps often contain expensive subscriptions, pay to unlock features, and excessive ads which all degrade user experience and make others inaccessible to the app. Pet's Mind, on the other hand, is completely free and easily accessible on both App Store and Play store. People who previously cannot afford expensive subscriptions can also enjoy the app and provide better care for their pets.

In the experiment section, we conducted a survey experiment to evaluate how well our application fits with a general audience. We created 10 questions rated on a 1–5 scale and asked 5 testers to use our program before completing the survey. The questions focused on usability, clarity of design, effectiveness of the AI Nutrition Expert, usefulness of the Health Tracking system, and overall satisfaction [7]. The most significant findings were that users rated navigation, AI responses, and ease of Health Tracking highly, with averages above 4.3 out of 5. The lowest ratings, closer to 4.0, came from design appeal and trust in AI accuracy. These results suggest that the app is both effective and user-friendly but can still improve in visual design and AI reliability. Overall, the experiment was successful because it confirmed that our app meets user needs while also highlighting concrete areas for future refinement.

2. CHALLENGES

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

2.1. Comprehensive Pet Tracking Made Simple

One potential skepticism is that our app doesn't allow easy tracking for the people's pet, which is not true. Our app allows the user to input any pets they have including their name, description, pet type, and also an image avatar of the pet. Moreover, the app allows the user to track detailed data about the pet, specifically the pet's weight, food intake, activeness, and age, which can be updated anytime. In addition, users can have multiple pets and manage multiple pets at once. This system allows the user to monitor their pet's well being and activity easily.

2.2. Real-Time AI Nutrition Guidance for Pets

Another potential concern is the lack of immediate responses to users' questions about their pets. Pet owners often don't want to wait minutes, hours, or even days for answers as this waiting period can be frustrating and stressful, particularly when owners are in a rush to resolve issues about their pets, our app features AI Nutrition Experts that provide real-time feedback, offering instant and reliable guidance whenever users need nutritional advice for their pets.

The AI Nutrition feature ensures that users receive instant, reliable, and personalized guidance without the delay, offering pet owners both confidence and convenience.

2.3. Building a Connected Pet Care Community

Another potential skepticism is that our app might feel more like a standalone tool instead of a community-driven platform. Our app does offer many tools for the pet owners like the pet tracker and AI Nutrition Experts. However, our app also contains a community forum page which every pet owner can post freely, sharing their experiences, tips, and stories on their pets. This fosters a friendly community which allows pet owners to connect with each other, exchange advice, and feel supported. By combining powerful tools for pet caring with an interactive forum, our app ensures that users not only receive personalized guidance but also experience a sense of belonging and shared purpose.

3. SOLUTION

The first screen upon opening Pet's Mind is the login/signup screen. The user should make an account using their email and make a custom password, if the user has an account and has signed in before, they will be automatically signed in upon opening the app.

After signing in, the users will see the Home Page which will allow them to navigate to the five different sections of the app: Home, Community Forum, AI Nutrition Expert, Health Tracking, and Profile [8].

The home page displays all the current pet of the user, along with buttons to navigate to other sections.

The community forum contains posts from other people sharing their stories, tips, and experience about their pets. Everyone can post on the community forum by clicking on the bottom on the bottom right of the page to create a post.

The AI nutrition page allows the user to ask questions about their pet and about pet nutrition and get a response real time. The page also provides some hint prompts for the user.

The Health Tracking page records all the information about the user's pet: weight, food intake, activeness, and age. Users can update their pet's information in the Health Tracking page and view their pet's status. On the bottom of the page there's also a food suggestion section which shows pet food suggestions for the user's pet.

Profile Page allows the user to update their user name, sign out, and delete accounts.

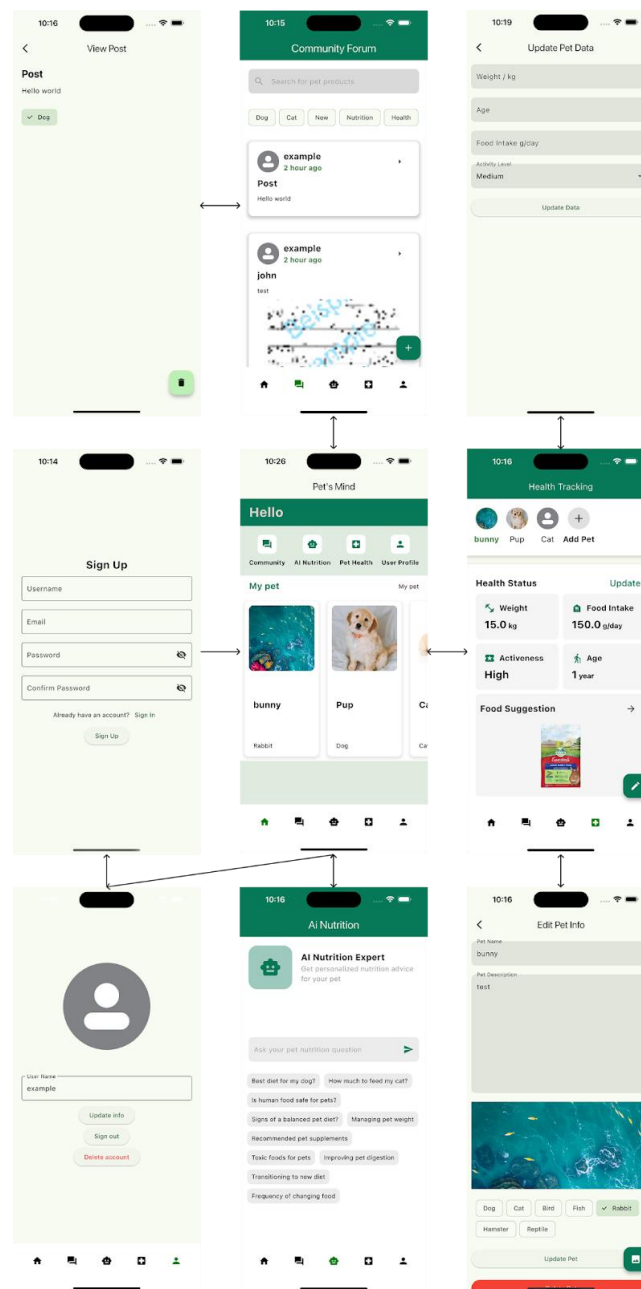


Figure 1. Overview of the solution

Pet's Mind contains an AI Nutrition Expert which allows the user to ask anything about their pet's nutrition and get an immediate response. This feature helps pet owners make informed decisions for their pets' diets. The component below shows the part where the user prompt is being passed to the Large Language Model, obtains a response, stores the data, and updates the UI [9].



```

1
2 List<MessageData> messages = [];
3 Future<void> onMessageSend(String message) async {
4   setState(() {
5     messages.add(
6       MessageData(
7         message: message,
8         isUserMessage: true,
9         createdAt: DateTime.now(),
10      ),
11    );
12  });
13   final content = [
14     ...messages.map((message) => Content.text(
15       ""
16       ${message.isUserMessage ? username : "AI"}
17       ${message.message}
18       ""
19     )),
20     Content.text(
21       "prompt",
22     ),
23   ];
24   final response = await model.generateContent(content);
25   setState(() {
26     messages.add(
27       MessageData(
28         message: response.text ?? "Sorry, I didn't understand that.",
29         isUserMessage: false,
30         createdAt: DateTime.now(),
31      ),
32    );
33  });
34 }
35
36 class MessageData {
37   final String message;
38   final bool isUserMessage;
39   final DateTime createdAt;
40
41   MessageData({
42     required this.message,
43     required this.isUserMessage,
44     required this.createdAt,
45   });
46 }

```

Figure 2. Screenshot of code 1

This section of code creates a chat interface for an AI nutrition assistant.

The code maintains a list of messages that stores the conversation history, where each message, modeled as the MessageData class, includes the text content, whether it came from the user or the AI, and when it was created.

When a user sends a message, the onMessageSend function first adds the user's message to the conversation list and updates the UI to show the user message for immediate feedback. Then it prepares a prompt for the AI model by creating a content array with the user's message along with a compiled conversation history as extra context [10]. The code calls an AI model to generate a response based on this input.

Once the AI responds, it adds the AI's reply to the message list as a MessageData with isUserMessage set to false. Then it calls the setState function again to update the UI. If the AI doesn't provide a valid response, it shows a fallback message saying "Sorry, I didn't understand that."

This creates a back-and-forth conversation flow where users can ask nutrition questions and receive AI-generated answers, with the entire conversation history preserved and displayed in the interface.

Our app contains a Community Forum page where every user can share anything about their pets. The page refreshes real time whenever someone adds a post, and also contains a search bar for

the user to search content inside the posts. Moreover, the page contains a filtering section below the searchbar, allowing the user to filter between different tags in the posts.

This code block shows the filtering logic and real time update logic of the page, which is crucial for improving user experience.

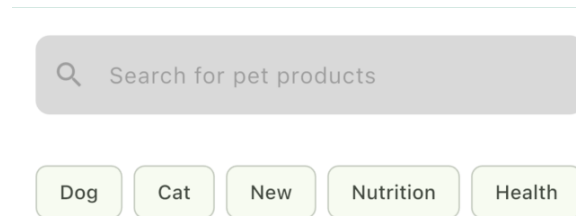


Figure 3. Screenshot of search box

```

1  StreamBuilder(
2    stream:
3      FirebaseFirestore.instance
4        .collection("Posts")
5        .orderBy("createdAt", descending: true)
6        .snapshots(),
7    builder: (
8      BuildContext context,
9      AsyncSnapshot<QuerySnapshot> snapshot,
10     ) {
11       if (snapshot.connectionState == ConnectionState.waiting) {
12         return Center(child: CircularProgressIndicator());
13       }
14       if (snapshot.hasError) {
15         return Center(child: Text("Error: ${snapshot.error}"));
16       }
17       if (!snapshot.hasData || snapshot.data!.docs.isEmpty) {
18         return Center(child: Text("No posts available"));
19       }
20       final posts = snapshot.data!.docs;
21       if (selectedCategories.isNotEmpty) {
22         posts.retainWhere((post) {
23           final postCategory = List<String>.from(
24             post["categories"] as List,
25           );
26           return postCategory.any(
27             (category) => selectedCategories.contains(category),
28           );
29         });
30       }
31       if (searchController.text.isNotEmpty) {
32         posts.retainWhere((post) {
33           final title = post["title"].toLowerCase();
34           final description = post["description"].toLowerCase();
35           final searchText = searchController.text.toLowerCase();
36           return title.contains(searchText) ||
37             description.contains(searchText);
38         });
39       }
40       return PostColumn(posts);
41     },
42   ),

```

Figure 4. Screenshot of code 2

This code creates a subscription that continuously monitors a collection of posts from Firebase Firestore.

It achieves real time updates by setting up a StreamBuilder that listens to the "Posts" collection. The query also automatically orders all posts by their creation date with the newest posts appearing first, improving user experience.

The builder function handles different states of the data stream. If the data is still loading, it shows a loading spinner; if something goes wrong, it displays an error message; if there are no posts in the database, it shows "No posts available".

Once the posts are retrieved, the code applies two filters to help users find relevant content.

First, if the user has selected specific categories, it filters posts to only show those that match at least one of the selected categories by checking if any of the post's categories overlap with the user's selections. Second, if the user has anything entered in the searchbar, it filters the posts by looking for the search text within both the post title and description, ignoring case differences to make searching more user-friendly.

Our app uses Firebase Authentication to handle accounts and user identity [14]. Besides Firebase Authentication, the app also contains a "users" collection to store user's profile data like user name and profile photo. Managing account changes in the app can be tricky and this code block shows how the app manages and updates account info whenever authentication changes.

A screenshot of a code editor showing a Java class named UserProvider. The class extends ChangeNotifier and has a DocumentSnapshot field named userData. It includes a constructor that sets up a listener for Firebase Authentication state changes. When a user signs in, it calls getAccountValue to fetch user data from the database. When a user signs out, it sets userData to null and notifies listeners. There is also an async method getAccount that takes a user ID and returns a DocumentSnapshot.

```
1 class UserProvider extends ChangeNotifier {
2     DocumentSnapshot? userData;
3
4     UserProvider() {
5         FirebaseAuth.instance.userChanges().listen((User? user) {
6             if (user != null) {
7                 getAccountValue();
8             } else {
9                 userData = null;
10            }
11            notifyListeners();
12        });
13    }
14
15    Future<void> getAccountValue() async {
16        if (FirebaseAuth.instance.currentUser != null) {
17            userData = await getAccount(FirebaseAuth.instance.currentUser!.uid);
18            notifyListeners();
19        }
20    }
21
22    Future<DocumentSnapshot?> getAccount(String uid) async {
23        try {
24            return await FirebaseFirestore.instance.collection("users").doc(uid).get();
25        } catch (e) {
26            print(e);
27            return null;
28        }
29    }
30 }
```

Figure 5. Screenshot of code 3

This code denotes a user management system that automatically keeps track of the current user's information in the app.

The system utilizes the provider package and creates a UserProvider class which extends ChangeNotifier, meaning it can notify other parts of the app whenever important data changes.

It sets up a listener that monitors Firebase authentication state changes which automatically triggers whenever a user signs in or out. When a user signs in, it calls getAccountValue to fetch the user's profile data from the database. If a user signs out, it clears the stored user data and notifies the app about this change.

The `getAccountValue` function checks if there's currently a signed-in user and, if so, retrieves their account information from the "users" collection in Firestore using their unique user ID. The separate `getAccount` function handles the actual database call to fetch user data. It also includes error handling that prints any issues to the console and returns null if something goes wrong. The `UserProvider` ensures that the app always has up-to-date information about who is currently using it, and notifies its consumers about authentication changes whenever it happens.

4. EXPERIMENT

For this experiment, we aim to test how well our program fits with a general audience of pet owners. By collecting survey responses, we can evaluate whether our app is easy to use, effective, and engaging. This is important because user satisfaction directly determines adoption and long-term success.

We designed a survey with 10 questions, each rated on a scale from 1 (strongly disagree) to 5 (strongly agree). These questions focus on usability, functionality, and satisfaction with our app's main features, including Health Tracking, AI Nutrition Expert, and the Community Forum. The questions were chosen to measure clarity of design, ease of navigation, usefulness of AI responses, and overall user experience. We asked 4–5 testers to interact with the app for several minutes before completing the survey. Their responses provide insights into strengths and areas needing improvement, ensuring our app meets real user expectations.

Survey Questions (1–5 scale):

The app was easy to navigate.

The design of the app felt clear and intuitive.

The AI Nutrition Expert gave helpful responses.

I found the Community Forum useful and engaging.

The Health Tracking feature was easy to understand and update.

I would use this app regularly to track my pet.

The app design felt visually appealing.

I trust the AI's responses to be accurate.

The app improved my knowledge about pet care.

I would recommend this app to other pet owners.

Question	Respondent 1	Respondent 2	Respondent 3	Respondent 4	Respondent 5	Average
1. Easy to navigate	5	4	5	4	5	4.6
2. Clear and intuitive design	4	4	5	3	4	4
3. AI gave helpful responses	5	4	4	4	5	4.4
4. Community Forum useful	4	4	5	4	4	4.2
5. Health Tracking easy	5	4	4	4	5	4.4
6. Would use regularly	5	4	4	4	5	4.4
7. Visually appealing design	4	3	4	4	4	3.8
8. Trust AI accuracy	4	4	3	4	4	3.8
9. Improved pet knowledge	5	4	4	4	5	4.4
10. Recommend to others	5	4	5	4	5	4.6

Figure 6. Table of experiment

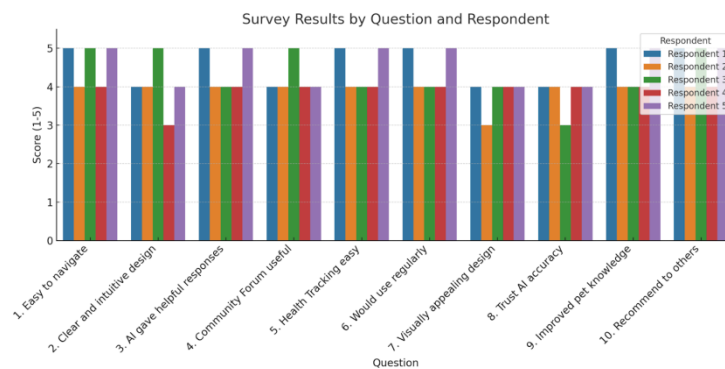


Figure 7. Figure of experiment

The survey results suggest that overall user satisfaction with our app is high. The average score across all questions was 4.3/5, indicating that most users found the app effective and easy to use. The strongest responses came from questions about navigation and helpfulness of the AI Nutrition Expert, which both averaged above 4.4. This shows that users particularly valued clear navigation and instant nutritional advice.

The lowest scores were in design aesthetics and trust in AI accuracy, averaging closer to 4.0. This suggests that while the app is functional, some users feel the interface could be more visually polished and the AI could improve in reliability.

Overall, the experiment demonstrates that the app successfully engages users and addresses their needs. The feedback provides valuable direction for future improvements, particularly in refining design appeal and strengthening AI response accuracy.

5. RELATED WORK

Diana's app is also a pet friendly mobile application that offers many tracking features like cut, health, feed, etc [11]. Both Diana's app and our app Pet's Mind allow the user to register/create pets and monitor their pets' activity and status by updating their pets' data. Compared to our app, Diana's app offers more detailed actions like showering, vaccination, deworming, etc. However, Diana's app does not have a place where pet owners can give and find tips and information about their pets. Our app Pet's Mind offers an AI Nutrition Expert which allows the user to obtain real time responses on any question they potentially have on their pet's diet. Moreover, our app also contains a community forum page where pet owners connect and support each other, fostering a sense of community.

Vanshri's app is an android app that serves as a pet care system. Vanshri's app is split into two sections: doctor's application and pet care application [12]. Pet owners can access pet training videos, diet information, along with doctors who can provide healthcare for their pets. However, different from our app, Vanshri's application does not offer a method for pet owners to track their pet's data like the pet's activeness, weight, food intake, etc. Our app also leverages AI to provide real time feedback to users' nutritional questions. Moreover, our app also serves as a community platform for pet owners to share their experiences and tips on raising their pets.

Dr. More's application is a web application that allows pet owners to register and track their pets [13]. One big difference is that Dr. More's application runs on the web instead of mobile, and uses HTML, CSS, and Django instead of our Flutter framework. Although applications on the web might offer broader reach, mobile applications do provide convenience for daily use.

Dr. More's app also offers tracking features and dietary consultations for pet owners. In comparison, our app not only offers real time answers for user's dietary concerns on their pet, but also builds community by providing a platform where pet owners can connect.

6. CONCLUSIONS

One limitation of Pet's Mind is its reliance on AI-generated responses for nutrition advice [15]. While the AI is helpful and quick, its answers are only as good as the data it was trained on, which means that occasional inaccuracies or overly general suggestions may occur. Some testers expressed that while the AI was useful, they would prefer more reliable sources linked directly to veterinary guidelines. In addition, although the app contains core functions like Health Tracking, AI Nutrition, and the Community Forum, there are still some missing features that could make the app more comprehensive, such as appointment scheduling with veterinarians, vaccination reminders, and offline functionality for areas with poor internet access.

Another limitation is that our user interface, while clear and easy to navigate, could benefit from a more polished and visually appealing design to make the experience more engaging. If given more time, we would refine the interface, test with a broader audience, and integrate advanced features such as push notifications and data visualization for pet health trends. In the future, we would also like to expand the AI's capabilities to provide multilingual support and integrate professional veterinary databases, ensuring more accurate and trustworthy guidance for pet owners.

REFERENCES

- [1] Cui, Yao, Zhaohui Zoey Jiang, and Qi Li. "Unlocking the value of real-time AI assistance: Who benefits, and why?." Qi, Unlocking the Value of Real-Time AI Assistance: Who Benefits, and Why (2025).
- [2] Pop, Mădălin-Dorin, and Andreas-Robert Stoia. "Improving the tourists experiences: Application of firebase and flutter technologies in mobile applications development process." 2021 International Conference Engineering Technologies and Computer Science (EnT). IEEE, 2021.
- [3] Chen, Yixing, and Maher Elshakankiri. "Implementation of an IoT based pet care system." 2020 fifth international conference on fog and Mobile edge computing (FMEC). IEEE, 2020.
- [4] Voronov, Sergey, et al. "AI meets real-time: Addressing real-world complexities in graph response-time analysis." 2021 IEEE Real-Time Systems Symposium (RTSS). IEEE, 2021.
- [5] Islam, Rashedul, Rofiqul Islam, and Tohidul Mazumder. "Mobile application and its global impact." International Journal of Engineering & Technology 10.6 (2010): 72-78.
- [6] Allen, Karen. "Are pets a healthy pleasure? The influence of pets on blood pressure." Current directions in psychological science 12.6 (2003): 236-239.
- [7] Glad, Jo Ann M., Sarah B. Kotchian, and Gerald M. Barron. "Developing a Local Comprehensive Environment and Health Tracking System." Journal of Environmental Health 66.10 (2004).
- [8] McCarthy, Danielle I. "Nutritional intelligence in the food system: Combining food, health, data and AI expertise." Nutrition Bulletin 50.1 (2025): 142-150.
- [9] Thirunavukarasu, Arun James, et al. "Large language models in medicine." Nature medicine 29.8 (2023): 1930-1940.
- [10] Tu, Xiaoguang, et al. "An overview of large AI models and their applications." Visual Intelligence 2.1 (2024): 34.
- [11] Andrade Arenas, Laberiano Matías, et al. "Design of a mobile application for the control of pet care." (2023).
- [12] Luayon, Alexis Anne A., et al. "PetCare: a smart pet care IoT mobile application." Proceedings of the 10th International Conference on E-Education, E-Business, E-Management and E-Learning. 2019.
- [13] Singh, Dr Pyar, Mr Sanket S. Gabhale, and Uma Chudaman Patil. "International Journal of Research Publication and Reviews." International Journal of Research Publication and Reviews (November 09, 2024) (2024).
- [14] 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.
- [15] Lee, Jae-woo, et al. "Development of AI-generated medical responses using the ChatGPT for cancer patients." Computer methods and programs in biomedicine 254 (2024): 108302.