Companion Bot using Flutter Framework

Sharun R[1],Sujitha K[2],Tamilarasi P[3],Vigneswaran S [4]
Computer Science and Engineering Department
KGiSL Institute of Technology,
Coimbatore,India

Abstract:

Mental health is an important issue in the world today. With a large population now working from home and staying away from loved ones, the mental health situation has deteriorated. As such, it becomes important to track and remedy any problems before they get too serious. A robot that supports a person's everyday life. Social robots range from slightly animated stuffed animals to intelligent android-like devices that function as real companions. Advanced social robots may be able to recognize family members and remind them of events. The project will involve the implementation of natural language processing (NLP) techniques to enable the bot to understand user input and respond appropriately. Additionally, the bot will utilize machine learning algorithms to learn from user interactions, adapt to their preferences, and provide increasingly personalized experiences over time.

Keywords: *Flutter, Java, Natural Language Processing*

I.Introduction

In recent years, the development of companion bots has gained significant attention due to their potential to provide personalized assistance and companionship to individuals. It offers a range of benefits, including personalized assistance, emotional support, and companionship. These bots have the potential to become integral parts of people's lives, providing comfort and engagement in various scenarios. In this context, this project aims to build a companion bot using the Flutter framework. Flutter, developed by Google, is a popular cross-platform development framework that enables the creation of high-performance, visually appealing user interfaces. It allows developers to build applications that can run seamlessly on multiple platforms, including Android, iOS, web, and desktop.

II.Methodology

The methodology for developing a Companion bots involves several key steps. First, Designing Conversational Flows involves creating a conversational flowchart or diagram to visualize the structure of the bot's conversations. Next, Natural Language Processing (NLP) identifying the NLP techniques and tools required for the bot to understand and generate natural language, and then Annotating the data to define intents, entities, and dialogue states. This step is crucial for supervised learning approaches or building training datasets for machine learning models. Implementing the dialogue logic by developing the logic that handles the bot's responses and generates appropriate replies based on user inputs. Finally Bot Development is done by Implementation of the companion bot using the chosen technology stack, such as Flutter framework.

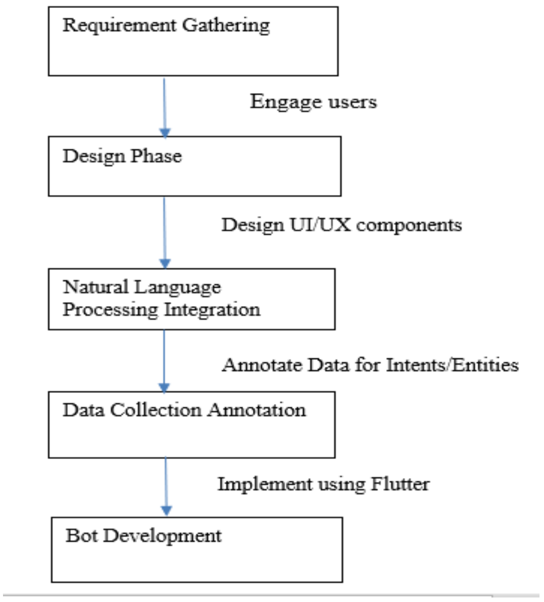


Figure1:System Architecture

III.Software Requirements

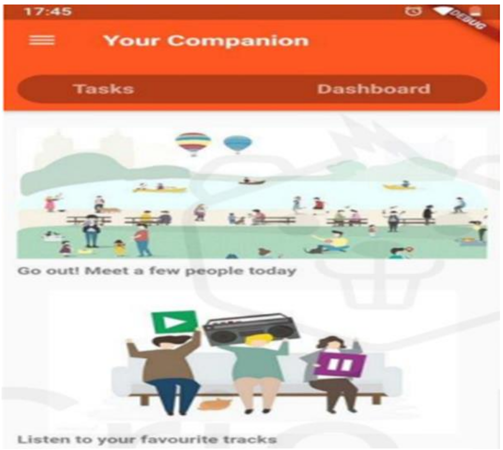
The software part in the system is divided into subparts based on functionality:-
The primary interface for building the user interface is Flutter.
The programming language for backend development system logic is Java.
For building the user interface screen, programming is done in Visual studio code.
For visualizing the output screen to user emulator is connected through the Android studio.

III.1. Functional Requirements

User Registration and Authentication: Implementing a mechanism for users to register, create accounts, and log in to the companion bot. This ensures personalized interactions and the ability to store user-specific information.
Conversational Interface: Developing a chat-based interface that allows users to interact with the companion bot through text input.

III.2.Non-Functional Requirements

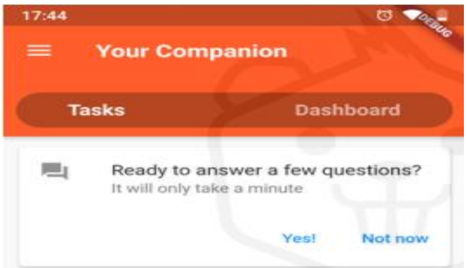
Performance: Ensures that the companion bot performs efficiently and provides fast responses to user queries. Minimize latency in processing user inputs and generating bot responses.
Scalability: Design the bot to handle increasing user loads and scale seamlessly as the user base grows. Consider factors like concurrent user interactions, data storage, and processing capacity.



Go out! Meet a few people today

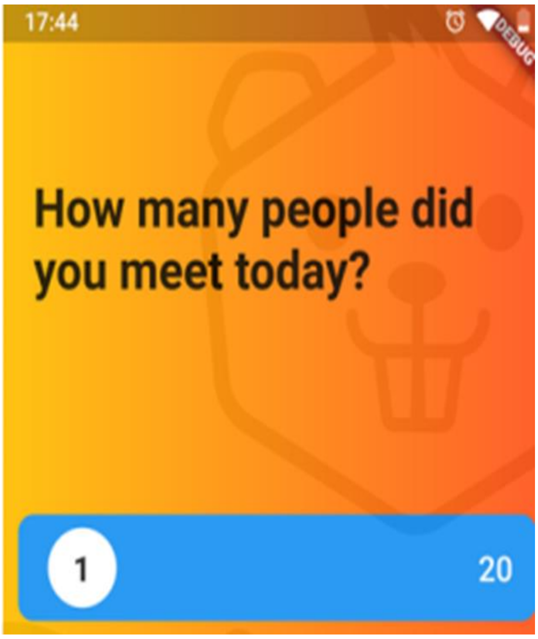
Listen to your favourite tracks

Figure 2: Main Screen



Ready to answer a few questions?
It will only take a minute

Figure 3: User Interface



How many people did you meet today?

Figure 4: Task Screen

V. Conclusion

Companion bot have provided empirical support for the necessity and designed in the manner that it care for mentally affected people. Our results demonstrate the stark differences in preferences and requirement between mentally affected people and bots, suggesting that engaging the end user in the design and development of companion robots is essential. Further desirable functions were also identified that are not currently included as standard on companion robots, such as eye-contact, life-simulation features, personalizing obeying commands and the potential for interactive language.

VI.Acknowledgement

WE ARE HIGHLY THANKFUL TO KGISL INSTITUTE OF TECHNOLOGY, COIMBATORE, INDIA FOR PROVIDING US THE FACILITY TO CARRY OUT PROJECT. WE ARE ALSO THANKFUL TO THE FACULTY MEMBERS (COMPUTER SCIENCE AND ENGINEERING DEPARTMENT), FOR THEIR CONTINUOUS ENCOURAGEMENT AND SUPPORT

Refrencence

1. Xiong, Caihua; Huang, Yongan; Xiong, Youlun (2008). Intelligent Robotics and Applications: First International Conference, ICIRA 2008 Wuhan, China, October 15-17, 2008

2. Sojka, Petr; Horak, Ales; Kopecek, Ivan (2008). Text, Speech and Dialogue: 11th International Conference

3. Wang, Kelly. iPal robot companion for China's lonely children. phys.org. Retrieved 10 December 2021.

4. Lamers, Maarten H. Verbeek, Fons J. (2011). Human-Robot Personal Relationships: Third International Conference, HRPR 2010, Leiden.

5. "NICOBO, a robot born out of empathy with consumers and sense of mission". Panasonic. Panasonic. Retrieved 2 May 2023