Voice-Controlled Smart Home Prototype to Assist an Elder in Home Care

Handy Wicaksono[1, a)], Petrus Santoso[1, b)], Indar Sugiarto[1, c)], Florenzo Dwipanjung[1]

Author Affiliations
[1]*Electrical Engineering Department, Petra Christian University*
Jl. Siwalankerto 121 – 131, Surabaya, Indonesia

Author Emails
[a)] Corresponding author: handy@petra.ac.id
[b)] petrus@petra.ac.id
[c)] indi@petra.ac.id

Abstract. This research aims to create a prototype of a smart home system to assist an elder in a home care scenario. Smart home system users can control devices such as lights using voice commands, keep track of the user's heart rate, monitor house situation, and store user's events using a schedule reminder. The smart home system will use Raspberry Pi 3 as the main microcontroller, Wemos D1 as a microcontroller on the smart home model, Broadlink RM Mini 3 as the medium to control an air conditioner and television, and Xiaomi Mi Band 2 as the wearable device. The program will use the Node-RED, NORA, and Firebase platforms which are very helpful in the development of the Internet of Things. System testing is done by testing the features of smart home systems, wearable devices, schedule reminders, as well as surveying the level of user satisfaction. The test results show that the smart home system can function properly. The microphone can detect voice commands up to a two-meter distance. Broadlink is able to control devices up to a three-meter distance. Wearable devices have succeeded in measuring the user's heart rate up to a distance of 9 meters, whereas the schedule reminder manages to store user events. Additionally, survey results indicate that respondents are satisfied with our system features.

INTRODUCTION

It is not cheap to attain paid formal care services offered by home care service providers or care centers. They are out of reach for most people, especially the underprivileged and those living with a tight budget, particularly elders. Many elders want to spend their retirement time in their own house [1]. However, modern-day home care services are no longer limited to elders; they now serve people of all ages. In the pandemic period, health services, such as home care, are needed more than ever. Therefore, there is a growing awareness to develop and implement cost-effective and efficient strategies and systems to provide health and monitoring services for people with limited access [2].

Previously, smart home automation was built upon embedded systems of Arduino Mega, ESP8266, and Raspberry Pi with the help of the MQTT protocol and the OpenHAB framework [3]. Arduino and OpenHAB programming allowed electronic devices to be controlled automatically using smart or manual modes. Communications between these embedded systems happened over the local network with the help of a wireless router. With this pre-designed system, users can control and monitor home electronic devices easily using a physical switch installed in their house and a physical wireless switch on the local network.

Another researcher attempted to develop a smart home automation system with Indonesian-based voice command [4]. His smart home automation used a Raspberry Pi as the brain of the system. To recognize the voice commands, the Julius Speech-to-text (STT) engine was used. The linguistic and acoustic models were made using the Hidden Markov Model (HMM) with the help of the Hidden Markov Model Toolkit (HTK) software.

A heart rate sensor using a KY-039 with phototransistors technology has been developed [5]. The heart rate sensor, along with other electronic components, was connected through an Arduino Nano with matching PINs. Next, they developed a program to detect heart rate through significant changes in the bloodstream as read by the sensor. The differences could be inferred by looking at changes in the infrared light intensity received by the phototransistor on the heart rate sensor. The heart rate record was then processed and converted into Beat per Minute (BPM) unit. Arduino microcontroller performed the processing and, if the heart rate matched a specific condition, it would then send a message to the router to send a text to a pre-registered number.

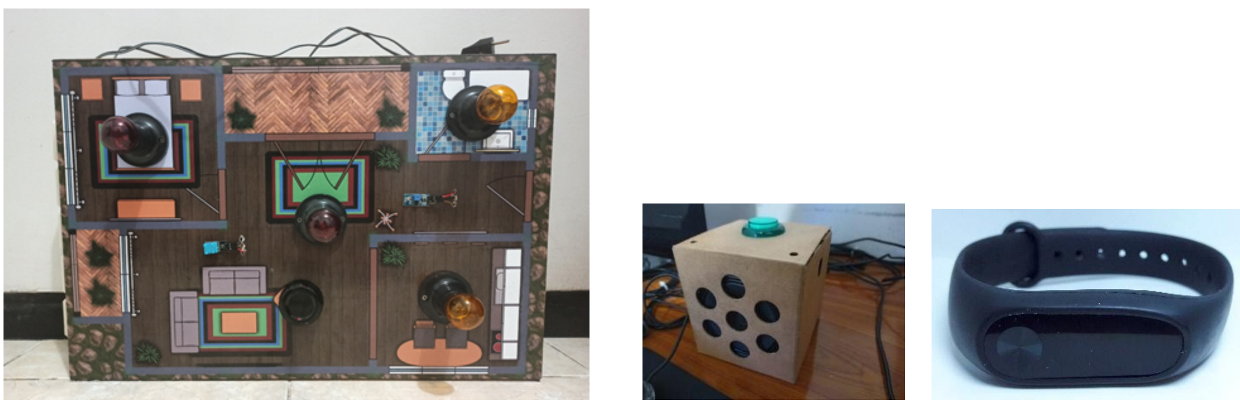
To continue the aforementioned studies, this research would like to further improve the integration of wearable devices in a smart home system. Heart rate reading results received from a wearable device, such as Xiaomi Mi Band 2, could be integrated into the smart home system and would then be turned into a parameter in that system that can be displayed on a dashboard. NodeRed will be used as the main tool to integrate components of our system, as it is has been proven effective in various applications [6].

RESEARCH METHOD

In the next subsections, we will describe the smart house prototype, the hardware design, and the software design.

The smart house prototype

The house miniature's size was a 60 × 40 × 15cm multiplex drawer with a detachable back (see Figure 1a.). This drawer was then used as a place to attach Wemos D1, Broadlink RM Mini 3, DHT 11 sensor, LDR sensor, light, and light switch. Meanwhile, the Google AIY Voice Kit will be placed separately from the model (see Figure 1b), and the Xiaomi MiBand 2 (see Figure 1c) will be worn by the elder.



(a) (b) (c)

FIGURE 1. The home miniature, Google AIY voice kit, and Xiaomi MiBand 2

Hardware Design

In creating this smart home hardware model, the author used Wemos D1 as the main microcontroller for the house miniature; LDR and DHT11 sensors as an environment sensor; light switches as manual controllers for the model lights; and a relay module that would control the lights based on the command signal from Wemos D1. In the voice kit hardware, the author used a Raspberry Pi 3 as the main microcontroller, Voice HAT accessory board as a HAT (Hardware Attached on Top) to attach the voice kit's buttons, speakers, and microphones, Voice HAT microphone board as a voice input receiver, Arcade-style push button as indicator lights, 3" speaker as a sound output, and Broadlink RM Mini 3 as an infrared controller for devices like TVs and ACs.

To connect Google AIY voice kit and the house miniature, NodeRed (with NORA library) is also installed in RaspberryPi 3. We can then develop a dashboard that can be accessed from the internet to monitor the house situation. Xiaomi MiBand 2 is connected to Raspberry Pi 3 via bluetooth. The heart rate reading is stored in a text file and then sent to NodeRed so it can be displayed on the dashboard. Figure 2 shows the block diagram of our system.

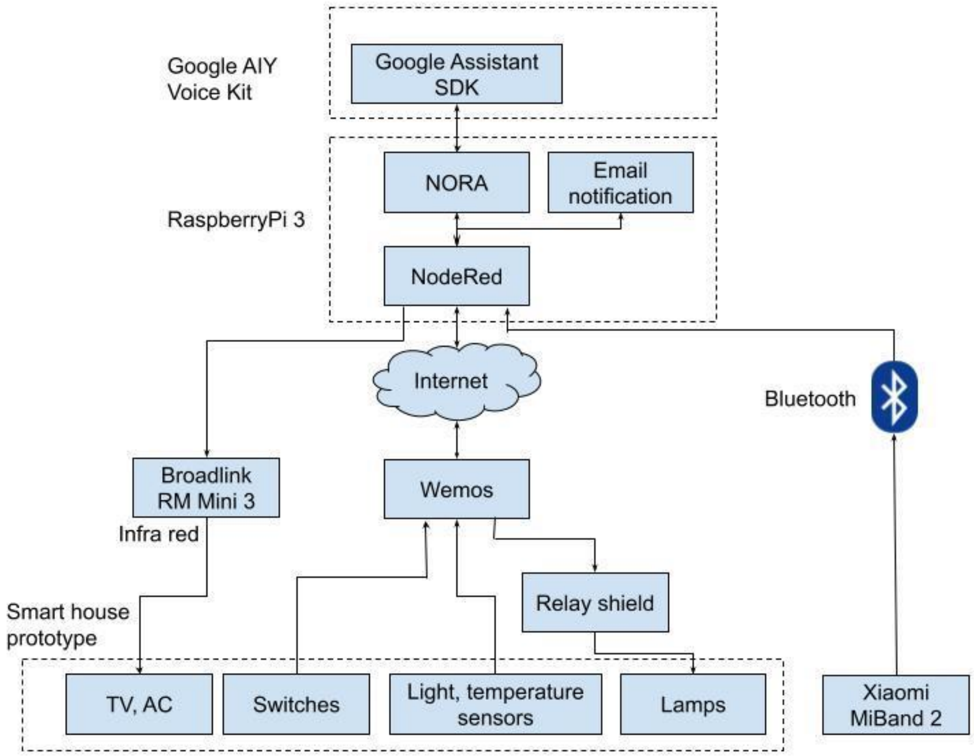


FIGURE 2. Block diagram of the smart home model (include Google AIY voice kit)

Software Design

Based on the preliminary survey results, here are the features that should exist in the smart home system:

1. Lights control manually (via application or via voice) and automatically
2. AC and TV control via voice
3. Temperature control
4. Track the heart rate and send alarm notification it is abnormal

We use web interface and voice as interaction mediums to our system. Voice can be useful for elders to interact with the system, as they might have difficulties in typing on the user interface. We use Google AIY voice kit to recognize the voice and use the acquired information to control the house. We use NORA (NodeRED Home Automation), a NodeRED library, so Google Assistant can send the acquired information to NodeRED that controls the smart house.

In our system, if the microphone detects a voice, then Google AIY kit will check whether it is one of the keywords: AC, TV, Lights, Sensor, or Schedule. Each keyword will lead to another keyword or phrase (e.g. turn on) that will perform a particular action (e.g. turn on a lamp). If there is no voice detected, then automatic temperature control and automatic lighting control are performed. Whenever the heart rate of an elder is too low or too high, an email notification will be sent. The complete flow chart of our system is shown in Figure 3.

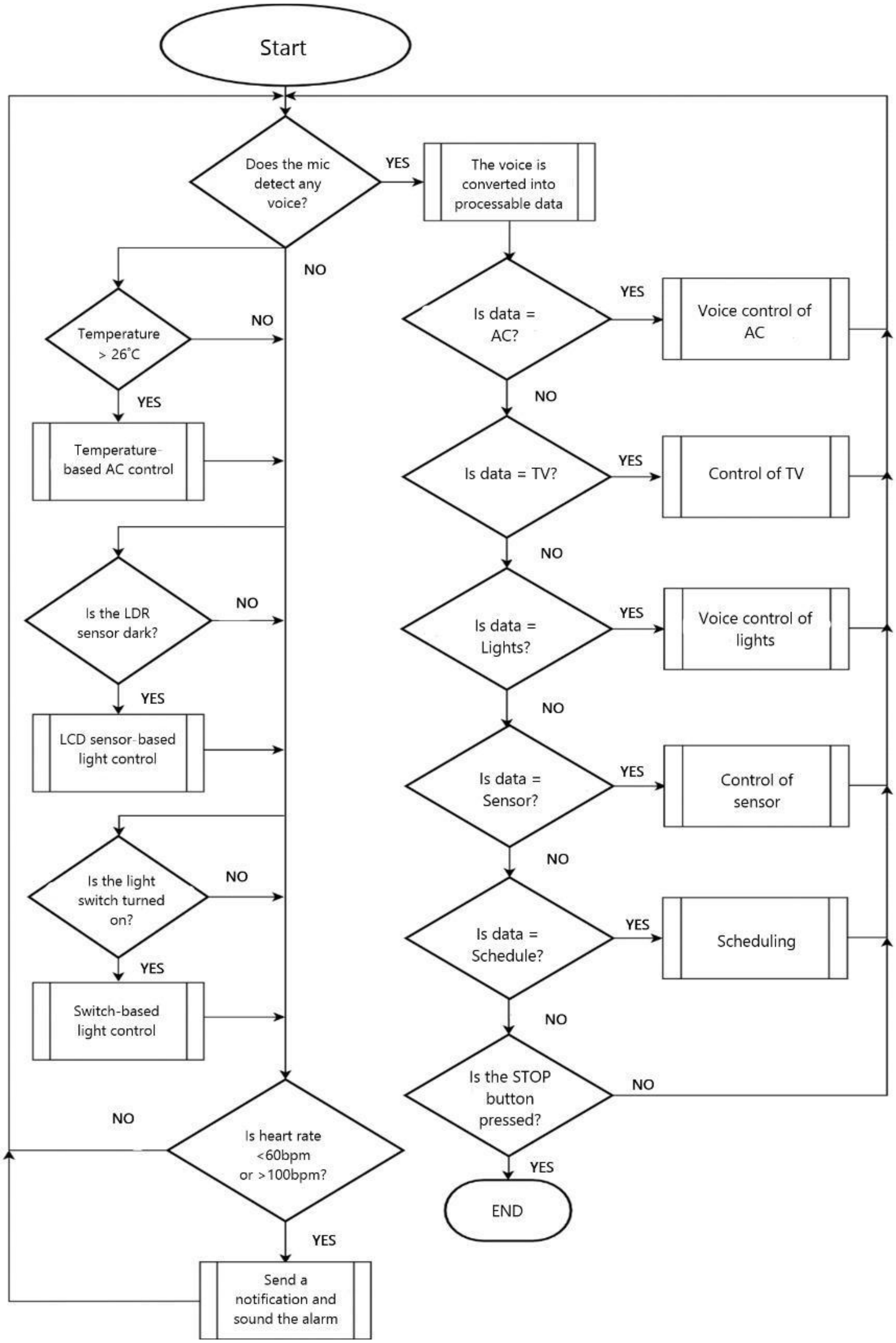


FIGURE 3. Flow chart of the prototype

Once NORA node receives data from Google Assistant, then it will be sent to Firebase. Wemos D1 or RaspbeeryPi then need to check available data in the Firebase. Figure 4 shows the flow of a voice-controlled light in NodeRed.

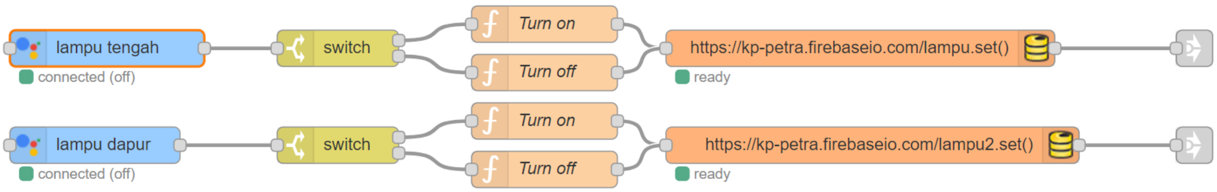


FIGURE 4. A sample of light control flow in NodeRed

Xiaomi Mi Band 2 will measure the patient's heart rate; then the data is sent and processed in the RaspberryPi. Later on, RaspberryPi will display the data and send the email notification if the heart rate is abnormal. Table 1 shows the heart rate of people based on their age.

The NodeRed flow that we use to process the heart rate data can be seen in Figure 5.

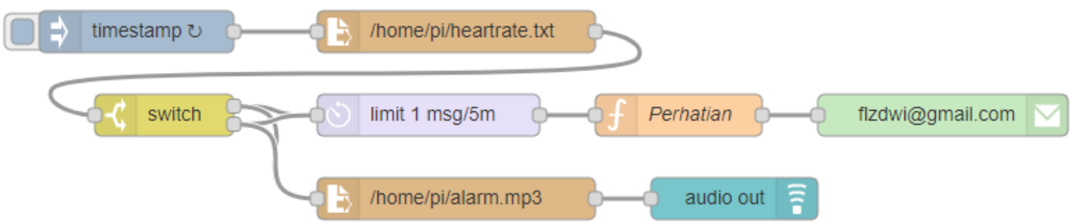


FIGURE 5. Node-RED flow for heart rate processing

RESULTS AND DISCUSSION

In this section, we will describe various experiments that are conducted to test the performance of our smart home prototype.

The smart house prototype experiment

1) Microphone Test Results

At this stage, the author tested the system by speaking verbal commands onto the microphone at various distances (1-5m). Five commands were tried for every distance. The voice commands, such as "turn off the main light" or "turn on the main light", were spoken onto the voice kit's microphone. Next to it, a smartphone was placed to measure the voice's decibel level and the room's silent ambience, from 1 meter and up. See Table 2 for the microphone test results.

TABLE 1. Target active heart rate zone based on age [7]

Age	Target active heart rate zone
20 years	100 – 170
30 years	95 – 162
35 years	93 – 157
40 yearsnode	90 – 153
45 years	88 – 149
50 years	85 – 145
55 years	83 – 140
60 years	80 – 136
65 years	78 – 132
70 years	75 – 128

TABLE 2. Microphone Test Results

Distance (m)	Average Sound Level (dB)	The success rate from 5 trials (%)
1	58,68	100
2	52,76	100
3	47,62	40
4	44,68	0
5	41,38	0

The test showed that the user had to be anywhere between 0-2m away from the microphone for the program to work as intended, and the user's voice must be louder than approximately 48 dB for the voice kit to detect the command.

2) Voice-Controlled Lights Test Results

At this stage, the author tested the voice-controlled lights on the smart home model. The purpose was to check whether the smart home's voice command feature responded promptly. This test was done by giving voice commands to the smart home system ten times and writing down the time it took to turn on or turn off the lights. The time recording started from the moment the voice command was given until the Wemos D1 received the data on Firebase (see Table 3). The duration was calculated from the timestamp difference from this process. The light status in the user interface and in the prototype can be seen in Figure 6.

Based on the test result, the last time it took for the smart home system to control the lights using the voice command was 1.75 seconds, whereas the longest was 2.68 seconds. The ten tests revealed that the average time it took to finish the process was 2.21 seconds.

3) Broadlink RM Mini 3 Test Results

At this stage, the author tested the Broadlink RM Mini 3's ability to control AC and TV. The purpose of this test was to check whether the Broadlink RM Mini 3 could still control the devices from various distances. It was done by controlling the AC and TV with the Broadlink RM Mini 3 on various distances (1-5m) five times on each distance and writing down the results. Table 4 shows the control test results of Broadlink RM Mini 2. According to these test results, the ideal distance between Broadlink and the target device should be between 1-3m for the program to work properly.

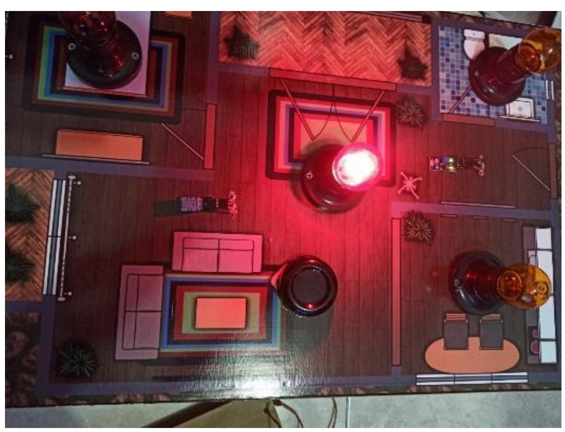
TABLE 3. Voice-controlled Lights Test Results

Attempt	Duration (in seconds)
1	2.42
2	2.11
3	2.57
4	1.84
5	1.97
6	2.29
7	1.75
8	2.43
9	2.68
10	2.07

TABLE 4. Broadlink RM Mini 3 Control Test Results

Distance	Amount of Attempts	Output
1 m	5	Success
2 m	5	Success
3 m	5	Success
4 m	5	Failure
5 m	5	Failure

(a)



(b)

FIGURE 6. Light status in the user interface and in the prototype

The wearable device experiments

At this stage, the author tested the data acquisition program used to monitor the user's heart rate on various distances (1-11m) five times on each distance. The testing process was done by running the data acquisition program used to monitor heart rate, with a 1-11m distance between the user and Raspberry Pi 3, both with and without a barrier. The results are shown in Table 5. Based on the test results, the ideal distance between the wearable device user and the Raspberry Pi should be between 0-9m without any barriers such as walls, or 0-8m when there are barriers in between, for the program to work properly.

The heart rate of two persons (in snapshot and trend) can be seen in Figure 7. If the heart rate is abnormal, then our system sends the notification email to the caregiver. Figure 8 depicts the notification email.

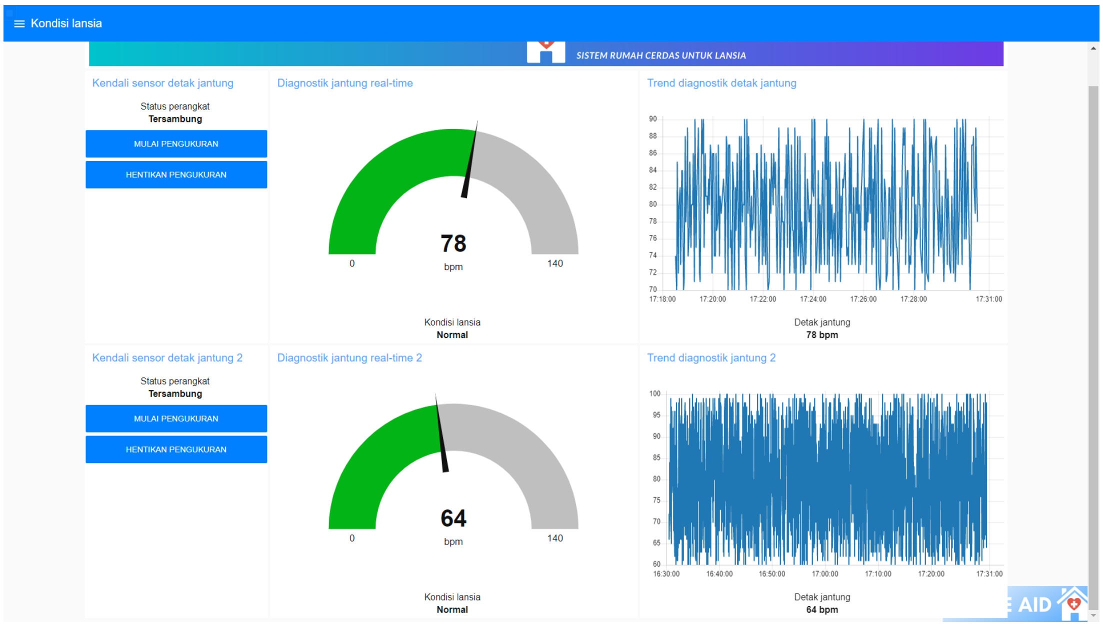


FIGURE 7. Heart rate snapshot and trend of two people

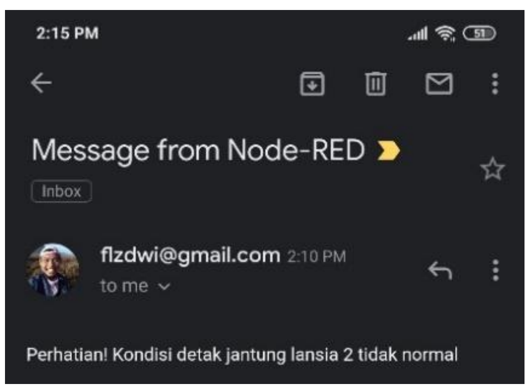


FIGURE 8. Email notification is sent when the heart rate is abnormal

TABLE 5. Wearable Device Test Results

Distance	Amount of Attempts	Successful Percentage	
		Without barrier	With barrier
1 m	5	100 %	100 %
2 m	5	100 %	100 %
3 m	5	100 %	100 %

4 m	5	100 %	100 %
5 m	5	100 %	100 %
6 m	5	100 %	100 %
7 m	5	100 %	100 %
8 m	5	100 %	100 %
9 m	5	100 %	40 %
10 m	5	60 %	0 %
11 m	5	0 %	0 %

The schedule reminder experiments

At this stage, the author tested the schedule reminder feature of the smart home system. The goal of this test was to see whether the smart home system could save a new schedule entry dictated through voice command. The test results showed that the smart home system had successfully displayed the new schedule entry on Google Calendar and the web dashboard (see Figure 9).

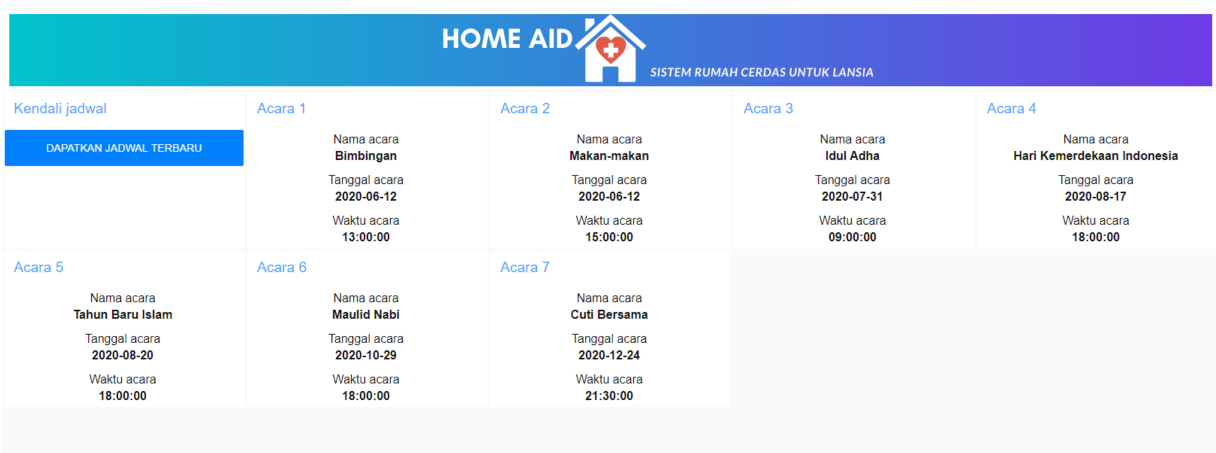


FIGURE 9. Schedule of a person in the web dashboard

User Satisfaction Survey

At this stage, the author conducted a survey to determine the level of user satisfaction regarding the smart home system. The target audiences were students or anyone who had a good relationship with their 60 years or older grandparents. Due to the pandemic, there were only ten respondents (Petra Christian University students) involved. Eleven items to measure user satisfaction are given to respondents. The survey result is shown in Table 6.

TABLE 6. User Satisfaction Survey Result

Number	Item	Unsatisfied	Less Satisfied	Neutral	Satisfied	Very Satisfied
1	Voice-controlled lighting feature	0 %	0 %	10 %	80 %	10 %
2	Voice-controlled AC feature	0 %	0 %	20 %	30 %	50 %
3	Voice-controlled television feature	0 %	0 %	30 %	50 %	20 %

4	Manual and automatic mode	0 %	10 %	40 %	30 %	20 %
5	Temperature and humidity monitoring	0 %	10 %	10 %	50 %	30 %
6	Heart rate monitoring using smartwatch	0 %	0 %	40 %	40 %	20 %
7	Events reminder feature	0 %	10 %	50 %	20 %	20 %
8	Web dashboard for system overview	0 %	0 %	30 %	30 %	40 %
9	Web dashboard for elders monitoring	0 %	10 %	20 %	30 %	40 %
10	Web dashboard for home monitoring	0 %	0 %	10 %	60 %	30 %
11	Web dashboard for schedules display	0 %	0 %	60 %	30 %	10 %

It is shown that most users are "satisfied" with our system. However, the second most frequent answers are "neutral", which means that the system still should be improved to meet user satisfaction.

CONCLUSIONS

1. The features of the smart home system function properly. The tests confirmed that users could control the smart home devices using voice commands. The ideal distance between the user and the microphone is no more than 2 meters because the microphone had difficulties picking up voice commands from a longer distance. On average, the smart home's voice control feature can complete the command in 2.21 seconds.
2. The wearable device and the smart home system are successfully integrated. The smart home system can monitor the user's heart rate in real-time as long as the heart rate monitor data acquisition program is running. The ideal distance between the wearable device user and the Raspberry Pi is 9 meters since the data acquisition program will not run if the distance is more than 9 meters.
3. The schedule reminder feature is successfully integrated within the smart home system. The Node-RED dashboard can be used to show the user's upcoming schedule.
4. The survey generally indicates positive responses, and most users are satisfied with our system.

ACKNOWLEDGMENTS

This research is partly funded by Hibah Penelitian Terapan Unggulan Perguruan Tinggi (Hibah PTUPT) 2021 from DRPM Kemristek/BRIN.

REFERENCES

1. J. V Hoof, G. Demiris, and E. J. M. Wouters, *Handbook of smart homes, health care and well-being* (Springer International Publishing, 2017).
2. S. Majumder, E. Aghayi, M. Noferesti, H. Memarzadeh-Tehran, T. Mondal, Z. Pang, and M. J. Deen, "Smart homes for elderly healthcare—Recent advances and research challenges" in *Sensors* vol. 17 (2017)
3. F. Hidayat, and I W. Mustika, "Perancangan Sistem Rumah Cerdas Berbasis Embedded Systems Menggunakan Framework MQTT dan openHAB", Skripsi S1 Teknik Elektro (Universitas Gadjah Mada, 2017).
4. R. Bogdan, P. Santoso, and R. Lim, "Kendali Suara Berbahasa Indonesia Untuk Automasi Rumah Tinggal", in *Jurnal Elektro Petra*, vol. 11(1), pp 1–6 (2018)
5. I. K. R. Arthana, I. M. A. Pradnyana, "Perancangan Alat Pendeteksi Detak Jantung Dan Notifikasi Melalui Sms", in *Seminar Nasional Riset Inovatif* (2017), pp. 889–895.
6. H. Wicaksono, P. Santoso, I. H. Putro, I. S. Hutomo, and P. Alvina, "Towards Integration of Heterogeneous Controllers in an IOT-based Automation System", in *E3S Web of Conferences* Vol. 188 (EDP Sciences, 2020).
7. UPMC HealthBeat, "What Is a Normal Heart Rate? - UPMC HealthBeat", (2015), https://share.upmc.com/2015/07/healthy-resting-heart-rate/