Smart Home Light Based Service Oriented Architecture and IoT

Yuda Irawan[1], Chrismondari[2], Yolnasdi[3], Ari Linarta[4], Muhardi[5], Anita Febriani[6]

[1]Department of Information System, STMIK Hang Tuah Pekanbaru, Pekanbaru, Indonesia
[2]Department of Electrical Engineering, STT Pekanbaru, Pekanbaru, Indonesia
[3]Department of Electrical Engineering, STT Pekanbaru, Pekanbaru, Indonesia
[4]Department of Information System, STMIK Dumai, Indonesia
[5,6]Department of Computer Science, STMIK Hang Tuah Pekanbaru, Pekanbaru, Indonesia

yudairawan89@gmail.com[1], chrismondari123@gmail.com[2], yolnasdi9@gmail.com[3], arie.linarta83@gmail.com[4] , muhardi@htp.ac.id[5], nitasuheri@gmail.com[6]

Abstract. The development of technology is very rapid in everyday life, so that it provides benefits for each. One of the advantages is that with technology, users can control electronic equipment which can be done with the smart home application (Smart Home) and from anywhere. This smart home application can help users control household electronic equipment such as lights, air conditioners, TVs, dispensers and others so as to reduce the waste of electricity when users forget to turn off household electronic equipment when traveling outside the house or anywhere. users are located. This research applies the concept of Service Oriented Architecture (SOA) and Internet of Things (IoT) on smart home light by integrating Ras-berry Pi with Python programming, web server (database) with PHP and An-droid programming with Java programming, and in experimental research that will be controlled. is the house light. The results of this research are in the form of a prototype of a home light controller using the IoT concept, web service and the Android SOA Smart home application with 7 buttons to control / monitor lights and this system is integrated with the concept of Service Oriented Architecture which is equipped with a token as system security. Tests have been carried out by controlling house lights remotely using a smartphone via the internet net-work and the results can be controlled properly.

1. Introduction

Smart Home is an application system that combines technology and services specific to the home environment with certain functions that aim to increase the efficiency, comfort and safety of its residents[1]. Smart Home usually consists of control, monitoring and automation of several home devices or appliances that can be accessed via a computer[2].

Internet of Things (IoT) is a concept that aims to expand the benefits of inter-net connectivity. Service Oriented Architecture (SOA) is a connected, standard based framework architecture that allows systems to integrate with each other data that was previously only tightly stored in various locations or

sources[3]. In other words, SOA is an architecture that supports business integration as a connected service and a pathway to innovation[4]. SOA is also a technology-based business strategy and the result of evolution to integrate multiple sources of information from different code sources or platforms[5].

Based on the explanation above, the research that will be carried out applies SOA and IoT concepts to Smart home light by integrating Raspberry Pi with Python programming, web server (database) with PHP and Android programming with Java programming that will be connected via the internet network[6]. This study aims to create a smart home system that can support multiple platforms via Android that can control and monitor devices at home via mobile[7]. In this study, the objective of this research is to create a smart home system that can support multiple platforms. In this study, what will be controlled is the lights in the house using a Raspberry Pi which is equipped with a modem and relay[8]. The lamp is connected to the Raspberry Pi via the GPIO (General Purpose Input Output) feature. Each lamp also has a relay that is installed between the GPIO and the lamp as a device ON / OFF switch[9].

2. Methodology

The research method is an important part of carrying out research, with the meth-od that has been determined by the researcher, the discussion of the problem and the research objectives are not out of context, because the researcher must follow the steps of the method[10]. The method used in making the system is the Prototype method. The following are the basic concepts of the research design carried out in Figure 1:
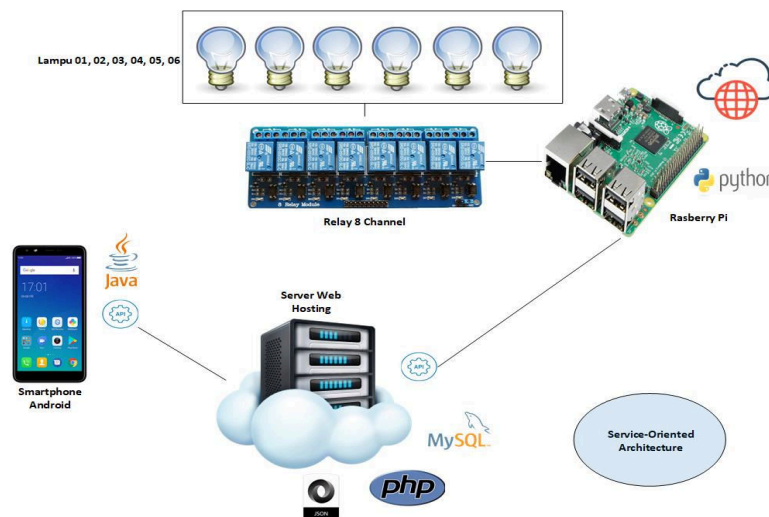

Figure 1. Research Design Concept

Light logic control is carried out by the 8 Channel Relay by providing logic 1 to activate the lamp and 0 to turn off the light. Only 6 ports are used because this pellet uses 6 lights and this relay is connected to the Raspberry Pi, where the function or task of the Raspberry Pi here is to continuously regulate to the web server by reading the port status on the GPIO on the web server . So that when there is a change in the request for the port status on the web server, the Raspberry Pi will send a change signal to the Relay, be it a value of 0 or a value of 1 as requested[11].

The reguest function can be done via Android. On Android, it has a function to monitor and control the lights installed on the home prototype by sending a port status change checklist to the database on the web server. Several stages in the process of making this prototype tool, it is at this stage that will determine the success of a hardware device[12].

The stages in the prototype model are as follows:
1. Data Collection

Researchers will collect data that will be managed as material for consideration or study in this study in order to make decisions at a later date[13].

2. Building Prototyping
   Design is done quickly and design represents all known aspects of hardware and this design forms the basis for the manufacture of the device in prototype form.

3. Evaluation of Prototyping
   Furthermore, after the prototyping development stage, the researcher will determine the format and needs of all devices, identify all requirements, and an outline of the system to be made.

4. System Coding
   At this stage the correct prototyping will be translated into programming in the form of an Android-based programming language.

5. Testing the Device
   At the testing stage of the device, the coding that has been previously made will be tested whether it can run well or there are still parts that need to be repaired or there are still parts that are not as expected.

6. Evaluation Tools
   Device evaluation is evaluating all the finished devices whether they are what they want or not. If not, the device will be revised again and returned to the previous stage. If the system has declared success, the system is ready to continue to the next stage.

7. Using the System
   This stage is the final stage of making a device using the Prototyping Model method. At this stage the software that has passed the test is ready to use.

3. Result and Discussion

The following is the Raspberry Pi and Relay design used in this study in Figure 2.
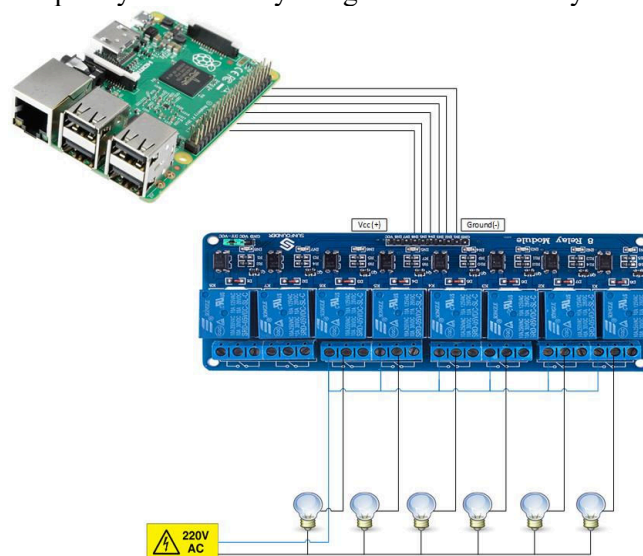

Figure 2. Raspberry Pi & Relay Circuit

In the picture above, several important components including 8 Channel Relays, Raspberry Pi, 5 watt light bulbs, cables, and 220V AC electric current. The ports used on the 8 Channel Relay are port 1, port 2, port 3, port 4, port 5 and port 6 which are connected to each lamp and so on are connected to an electric current with a 220V AC setting. While at the top of the 8 Channel Relay, there are 10 pins with 8 pin relay ports, 1 pin for Vcc (+) and 1 pin for grounding. The pins connected to the Raspberry Pi are pin 1, pin 2, pin 3, pin 4, pin 5 and pin 6.

The results of the study with the title "Smart Home Based Service Oriented Architecture" are implemented in accordance with the design. The results can be seen in Figure below.

Figure 3. Implementation Results of prototype house lights controlled from Android

In Figure 3, you can see the results of controlling prototype house lights using android with the SOA Smart home application. When in the default condition (lights off), light 1 is pressed on Android, room 1 light will turn on and the light button dis-play in the application changes from red to green. And so on in accordance with the designation of each button. the results of the prototype or prototype from the house. There are 6 lamps with each room. Lamp 1 is in room 1, lamp 2 is in room 2, lamp 3 is in the middle room, lamp 4 is in the bathroom, lamp 5 is in the garage and lamp 6 is on the house teras. In room 1, there is 1 small light that you get called an indicator light which functions to find out the connection of the Raspberry Pi. When the indicator light flashes continuously, it means that the Raspberry Pi is not connected to the internet and when the indicator light flashes every few seconds, then you can be sure that the Raspberry Pi is connected to the internet. Then there is an outlet to turn on the Raspberry Pi device.

Raspberry Pi circuit connected to the 8 Channel Relay then from the 8 Channel Relay connected to the lamps with a 220V AC source. The port used on the Raspberry Pi is GPIO port 16 for lamp 1 or room lamp 1, GPIO port 20 for lamp 2 or room lamp 2, GPIO port 21 for lamp 3 or living room lamp, GPIO port 13 for lamp 4 or room lamp shower, gpio port 19 for lamp 5 or gar-age light and gpio port 26 for lamp 6 or patio light. Whereas in Relay 8 Channel port 1 is used for lamp 1, port 2 is used for lamp 2, port 3 is used for lamp 3, port 4 is used for lamp 4, port 5 is used for lamp 5, port 6 is used for lamp 6, port 7 is empty and port 8 is empty.

The following are the results of the SOA Smart home application along with the explanation which can be seen in the pictures below.

Figure 4. Display Lamp All On

In Figure 4 the light button on the application is green because the status of the light is currently on or on. The big green button in the image above functions to turn the lights on or off at once for all rooms. The testing in this study is testing the time to access the web server with certain upload and download limits. This test aims to find out how long it takes to access the server from the client with limits limited to 1 Kbps, 4 Kbps, 8 Kbps, 16 Kbps, 32 Kbps, 64 Kbps, 256 Kbps, 512 Kbps and unlimited. The following is a sample test with a download and upload limit of 64 Kbps.

Testing was carried out by accessing the web server from the Raspberry Pi and sharing the limits using the mikrotik RB941-2nD. In Table 3 it can be seen that the test with a download and upload limit of 64 Kbps was carried out 20 times with the number of requests as much as 2 data and the average size on the web server that was loaded was 416.80 Bytes with an average time of 0.39 seconds.

4. Conclusion

In this research with the title Smart Home Based on Service Oriented Architecture and Internet of Things, system testing has been carried out so that the following conclusions can be drawn: The results of the study are in the form of home prototypes, web services and SOA Smart home applications that are integrated with the concept of Service Oriented Architecture. This system is equipped with a token as its security, so that in testing the level of system security, results are obtained as desired (valid) by testing without a token, wrong token and correct token when accessing the system. The application of the SOA concept in Smart Home certainly uses the internet, so that testing of internet data usage limits and how long it takes to access the web ser-vice from the client with the download / upload speed has been tested. As for testing 20 times with a download / upload limit of 1 Kbps, it took an average of 11.98 seconds to access the web service with an average size of 459.75 Bytes, a download / upload limit of 4 Kbps took an average of 4.58 seconds. To access a web service with an average size of 426.45 Bytes, a download / upload limit of 8 Kbps takes an average of 2.98 seconds to access a web service with an average size of 424.45 Bytes, a download / upload limit of 16 Kbps requires average time 0.67 seconds to access the web service with an average size of 418.3 Bytes, download / upload limit 32 Kbps takes an average of 0.42 seconds to access the web service with an average size of 408.6 Bytes , the 64 Kbps download / upload limit takes an average of 0.39 seconds to access the web service with an average size of 416.8 Bytes, the 256 Kbps down-load / upload limit takes an average of 0.37 seconds to access the web service with the average size is 406.3 Bytes, the download / upload limit of 512 Kbps takes time an average of 0.35 seconds to access a web service with an average size of 408.6 Bytes, unlimited download / upload limits of Kbps require an average of 0.25 seconds to access a web service with an average size of 406.3 Bytes.

References

[1]   A. Amato and A. Coronato, "An IoT-Aware architecture for smart healthcare coaching systems," *Proc. - Int. Conf. Adv. Inf. Netw. Appl. AINA*, pp. 1027–1034, 2017, doi: 10.1109/AINA.2017.128.

[2]   H. Q. Al-Shammari, A. Q. Lawey, T. E. H. El-Gorashi, and J. M. H. Elmirghani, "Resilient Service Embedding in IoT Networks," *IEEE Access*, vol. 8, pp. 123571–123584, 2020, doi: 10.1109/ACCESS.2020.3005936.

[3]   M. E. P. Hernandez and S. Reiff-Marganiec, "Autonomous and self controlling smart objects for the future Internet," *Proc. - 2015 Int. Conf. Futur. Internet Things Cloud, FiCloud 2015 2015 Int. Conf. Open Big Data, OBD 2015*, pp. 301–308, 2015, doi: 10.1109/FiCloud.2015.89.

[4]   S. Ahmed, M. Ilyas, and M. Yasin Akhtar Raja, "Smart living: Ubiquitous services powered by ambient intelligence (AmI)," *HONET-ICT 2019 - IEEE 16th Int. Conf. Smart Cities Improv. Qual. Life using ICT, IoT AI*, pp. 43–48, 2019, doi: 10.1109/HONET.2019.8908054.

[5]   I. A. F. Anto, A. Munandar, and M. I. Rizqyawan, "Modeling with SoaML Applied for Smart Room Based on Internet of Things," *2019 Int. Conf. Comput. Control. Informatics its Appl. Emerg. Trends Big Data Artif. Intell. IC3INA 2019*, pp. 125–128, 2019, doi: 10.1109/IC3INA48034.2019.8949610.

[6]   Y. Eshetu, P. Bhuyan, and A. Behura, "A service-oriented IoT system to support individuals with ASD," *Proc. 2018 Int. Conf. Commun. Comput. Internet Things, IC3IoT 2018*, pp. 251–256, 2019, doi: 10.1109/IC3IoT.2018.8668138.

[7]   L. Ren, X. Zhang, J. Wang, S. Tang, and N. Gong, "Design of hospital beds center management information system based on HIS," *Proc. - 2017 IEEE Int. Conf. Bioinforma. Biomed. BIBM 2017*, vol. 2017-Janua, pp. 1093–1096, 2017, doi: 10.1109/BIBM.2017.8217808.

[8]   P. Pannuto, P. Dutta, W. Wang, and B. Campbell, "A Modular and Adaptive Architecture for Building Applications with Connected Devices," *Proc. - 2018 IEEE Int. Conf. Ind. Internet, ICII 2018*, no. Icii, pp. 1–12, 2018, doi: 10.1109/ICII.2018.00009.

[9]   B. Omoniwa, R. Hussain, M. A. Javed, S. H. Bouk, and S. A. Malik, "Fog/edge computing-based IoT (FECIoT): Architecture, applications, and research issues," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4118–4149, 2019, doi: 10.1109/JIOT.2018.2875544.

[10]   Y. Irawan and R. Wahyuni, "Electronic Equipment Control System for Households by using Android Based on IoT (Internet of Things)," *J. Phys. Conf. Ser.*, vol. 1783, no. 1, p. 012094, 2021, doi: 10.1088/1742-6596/1783/1/012094.

[11]   A. Wall, V. Altmann, J. Müller, H. Raddatz, and D. Timmermann, "Decentralized configuration of embedded web services for smart home applications," *11th Annu. IEEE Int. Syst. Conf. SysCon 2017 - Proc.*, 2017, doi: 10.1109/SYSCON.2017.7934759.

[12]   C. L. Hu, S. Chen, L. Guo, C. Chootong, and L. Hui, "Home care with IoT support: Architecture design and functionality," *Ubi-Media 2017 - Proc. 10th Int. Conf. Ubi-Media Comput. Work. with 4th Int. Work. Adv. E-Learning 1st Int. Work. Multimed. IoT Networks, Syst. Appl.*, 2017, doi: 10.1109/UMEDIA.2017.8074155.

[13]   H. Fonda, Yulanda, M. Ikhsanudin, Muhardi, and Y. Irawan, "Application of Certainty Factor Method to Identify Pests in Crystal Jamboo Plants Application of Certainty Factor Method to Identify Pests in," *J. Phys. Conf. Ser.*, vol. 1783, no. 1, p. 012053, 2021, doi: 10.1088/1742-6596/1783/1/012053.