

PLO User Interface based on Telegram Bot

Bambang Nurcahyo Prastowo*¹, Nur Achmad Sulisty Putro², Oktaf Agni Dhewa³

^{1,2}Department of Computer Science and Electronics, FMIPA, UGM, Yogyakarta

³Master of Computer Science, FMIPA UGM, Yogyakarta

e-mail: *prastowo@ugm.ac.id, nur.achmad.s.p@mail.ugm.ac.id, oktaf.ad@mail.ugm.ac.id

Abstrak

Penggunaan sistem instant messaging telah melekat lengkap dengan sistem notifikasinya pada pengguna sebagaimana sistem telepon dan SMS. Telegram adalah satu-satunya sistem instant messaging populer yang membuka source code program aplikasinya serta menyediakan API bagi pengguna untuk mengembangkan sistem bot yang memungkinkan kita memanfaatkan aplikasi instant messaging untuk akses sistem informasi. Penelitian ini mempelajari seberapa jauh telegram bot dapat dimanfaatkan sebagai antarmuka pengguna akses ke sistem paperless office. Dengan keterbatasan arus pesan tunggal per pengguna, sistem tukar pesan Telegram dapat dimanfaatkan sebagai antarmuka pengguna ke sistem paperless office. Sifat multi arus pesan di PLO dapat diatasi dengan memanfaatkan fasilitas reply ke pesan dalam arus yang akan dikomentari lebih lanjut.

Kata kunci— sistem komunikasi, media sosial, pesan instan, integrasi sistem.

Abstract

Instant messaging services usually integrate a notification system on their users' devices, as phone calls and short message service (SMS) systems do. Telegram is – as far as we are aware – the only popular instant messaging service that uses open source code. Telegram also provides APIs for their users, enabling the development of a bot system that allows instant messaging application to access information. Here, we study the potential use of Telegram bot as a user interface to a paperless office (PLO) system developed in our institution. We found that Telegram bot improves communication among the users; however, as the amount of messages increased, the server becomes overloaded. This limitation suggests that future works need to be directed to improving the efficiency of the bot.

Keywords— communication system, social media, instant messaging, system integration.

1. INTRODUCTION

The advancement of communication technology, including the widespread use of telephones and smartphones, has a major impact in shifting social interaction behaviors. Personal and group-chat facilities on smartphones do not only facilitate communications within a community, but also change the style of communications and coordinations [1]. These facilities have been widely used in various settings including universities [2].

An internal institutional study conducted by Ahmad and Orton (2010) [3] at IBM shows that the increase of smartphone use in working environment for accessing e-Learning materials did not improve the performance of the employees. In fact, the use of smartphones increasingly dominates the active hours of professional workers and increases work-home interference [4].

Therefore, some strategies should be implemented so the use of smartphones could improve or at least maintain employee performance.

The biggest contributors to the high smartphone use are instant messaging services and Voice over IP (VoIP) [5], with Whatsapp as the most widely used application [6]. Similar to WhatsApp is Telegram, which in less than 3 years has gotten more than 100 million active users [7]. An interesting thing about Telegram is the availability of a client source code that can be assembled by the users. Therefore, it is possible to integrate the Telegram application with an information system developed separately. In addition to providing client source code, Telegram also provides an API for bot development, a program system designed with a simple query-response interface that can be integrated as a user. Using the bot, the other users can send question messages to get responses from related bot systems [8].

An in-house paperless office (PLO) web-based application has been developed by PPTIK (now DSSDI) Universitas Gadjah Mada (UGM), and implemented in various units for different purposes. Aside from being an office communication system, PLO has also been applied as a general social communication portal. PLO formal specifications are available in the literature [9]. Meanwhile, the API access to the PLO system has been developed in the `contact.ugm.ac.id` system and published [10].

TechCrunch technology information portal launches new services in the form of Telegram Bot that can be used by users to dig up user interface information in the form of instant messaging [11]. This service utilizes the bot's ability to send custom keyboards that are used for navigating information in the system. The online Assist service portal system uses Telegram Bot to connect users with on-line service booking systems such as Uber and OpenTable [12]. The trust level of the Telegram system is quite high as evidenced by the Sberber Bank in Russia venturing to launch bot-based telecommunication applications [13]. Some previous studies support the claim that Telegram is superior to other instant messaging services (e.g., WhatsApp, Viber), in many ways: it is able to synchronize with other applications, has super fast services, provides reliable backup source, and has better security features [14].

Chat bot has also been applied for patient health service communication system. Currently, communications in health care services are based on e-mail, and lately mobile messengers are being developed, integrating mobile health and mobile messenger. This kind of system utilizes chat features as in smartphone applications. Chat bot as a communication device enables the reduction of costs and streamline the patient's operational time in obtaining consultation services [15]. Telegram has also been successfully used in e-learning systems in the field of urology [16], to improve practical competence for medical clinic graduates. This particular system integrates two bot telegram and can be accessed through various devices.

Telegram API is a feature that allows developers and users to create programs on the messaging interface component. Telegram API also allows developers and users to build their own Telegram client. In this study, we use this API to access the PLO system. We aimed to study user behaviors upon the integration of Telegram bot with PLO. Our observations revealed that the bot improves communications among users. We also found that the increase of message flow could overload the server; therefore, other strategies need to be designed and implemented to increase the efficiency of the bot.

2. METHODS

Our research methodology was applied directly to the PLO system used in the Faculty of Mathematics and Natural Sciences (MIPA) UGM, and was conducted as the following:

1. Formulating PLO chatter (sub comment) in the form of instant messaging
2. Designing PLO system for accessing telegram bot
3. Developing Telegram chat models to facilitate sub comment discussions
4. Observing user behavior
5. Analysing the advantages of Telegram chat archiving.

The development of the new system in this study did not change how the original installed system operates. The Telegram bot is integrated to the archive module which is a component used mandatorily by all faculty staff. Telegram has two different ways to receive bot messages with information on the PLO, i.e., long polling and WebHook. Here, we used WebHook that is placed directly on the server of PLO MIPA. The flow of information from PLO server to users through Telegram bot is illustrated in Figure 1.

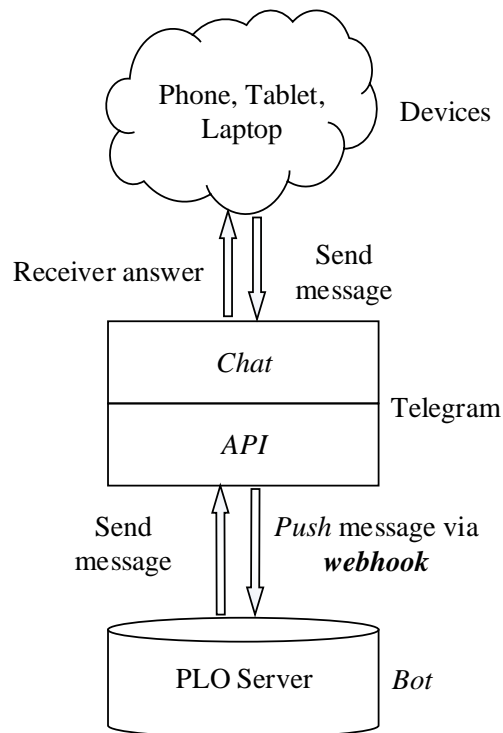


Figure 1 Flow of information from PLO server to users through Telegram bot

The choice of WebHook is based on real time communication to match the bot characteristics. The bot is developed in a pure PHP environment without an embedded framework in the PLO module. PHP is a script programming language with interpreted server-side that has continuously updated [17]. The bot calls the API provided by Telegram server directly without an intermediary module developed by a third party. The API that access to the PLO system has previously been developed in `contact.ugm.ac.id` system [10]. User Interface built using standard Telegram client can be accessed from devices which support the operation of the client.

This research was carried out in MIPA UGM; bot implementation is embedded directly in the PLO-MIPA system by taking into account all aspects of prudence in the development of information systems.

3. RESULTS AND DISCUSSION

The results of this study are a design and implementation of telegram bot that embedded in the operated system at Faculty of Mathematics and Natural Sciences, UGM. This bot system become a service access agents for the paperless office system.

3.1 System Design

The design of user interfaces for PLO using Telegram Bot is outlined in User Case and Activity diagrams. It follows the discipline of software design using UML. The selection of UML design because it can accommodate various software development models, especially object-based software and provide an easy-to-understand interface diagram [18].

3.1.1 Use case

There are two access points for using the system from the user side: Telegram Bot connection with PLO system and chat session with Telegram Bot for PLO access. Connecting with the PLO system is done by requesting an activation code. The code is generated on the PLO Web. The activation code is used to connect bots with a PLO account by sending a message / reg <activation code> to a bot telegram named PloFMIPAUGMBot. This relationship is presented in Figure 2.

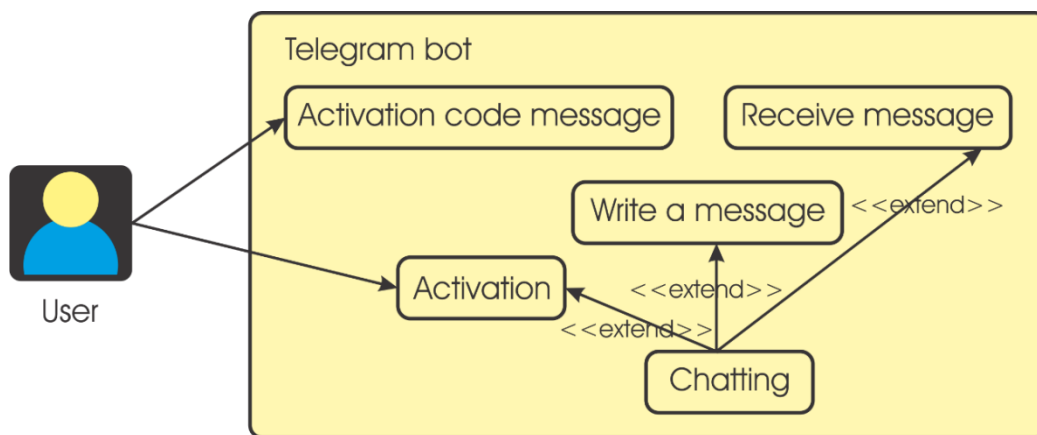


Figure 2 User connection using telegram bot

After the activation process is successful, then the user can interact with the PLO server through the chat interface by the relevant bot, i.e. write the message and receive the message. Message reception is passive, it means that it is carried out by the "coercion" mechanism of the message entering the chat chain of the user by the server.

3.1.2 Activity

There are four activities that can be carried out by the built-in bot system, including activation of the bot as a PLO interface, receiving PLO messages, sending PLO messages and reading old messages in the PLO.

Bot activity as a PLO interface

In the PLO, bot's activity as an user interface involves 4 communication agents, i.e. :

1. PLO users in the Web Browser,
The target of PLO users in the Web Browser is the staff of Universitas Gadjah Mada, especially when it is run in the environment of the Faculty of Mathematics and Natural Sciences, both staff who act as lecturers and educators.
2. PLO Server,
PLO Server will be connected directly to the data base as an archiving media for information messages. These messages are sent via the web by users involved in the system.
3. Telegram Bot, and
4. PLO users at Telegram.

To be able to use the bots as the PLO interface system, users must activate their account by requesting an activation code to the server via the web. When the web system has given an activation code, the user is asked to send the code to the bot's telegram via the reg command or register as synchronizing the PLO system with a telegram. The illustration of the process is presented in Figure 3.

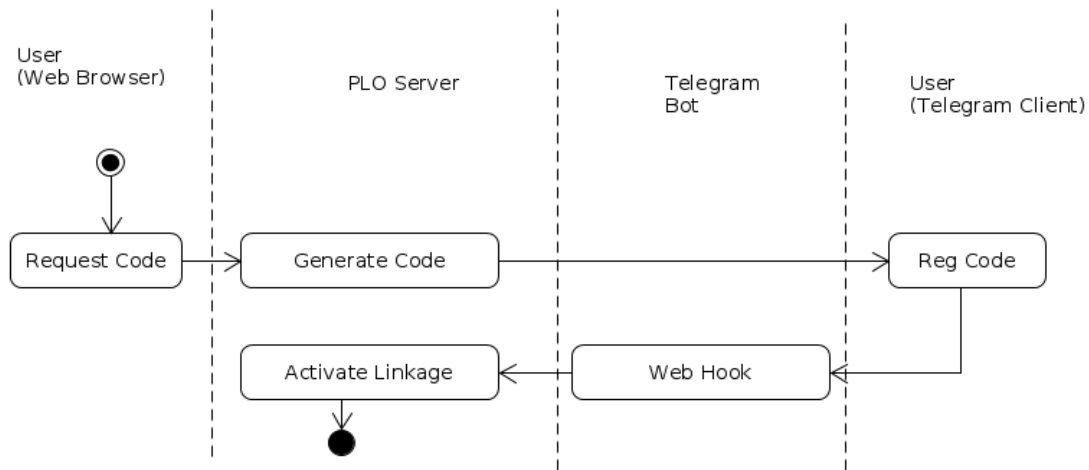


Figure 3 Telegram Bot Relationships with PLO User Login

Furthermore, chat to the telegram bot will be connected directly to the PLO with a session on behalf of the user who requested the activation code.

Getting PLO messages

In complete implementation, all messages for each user will be sent directly to the chat user who is concerned with the PLO interface bot. The sending data process is carried out by a notification module as shown in Figure 4. Basically all communication carried out by the Archive module in the PLO is formal office communication, so all of its users must pay full attention.

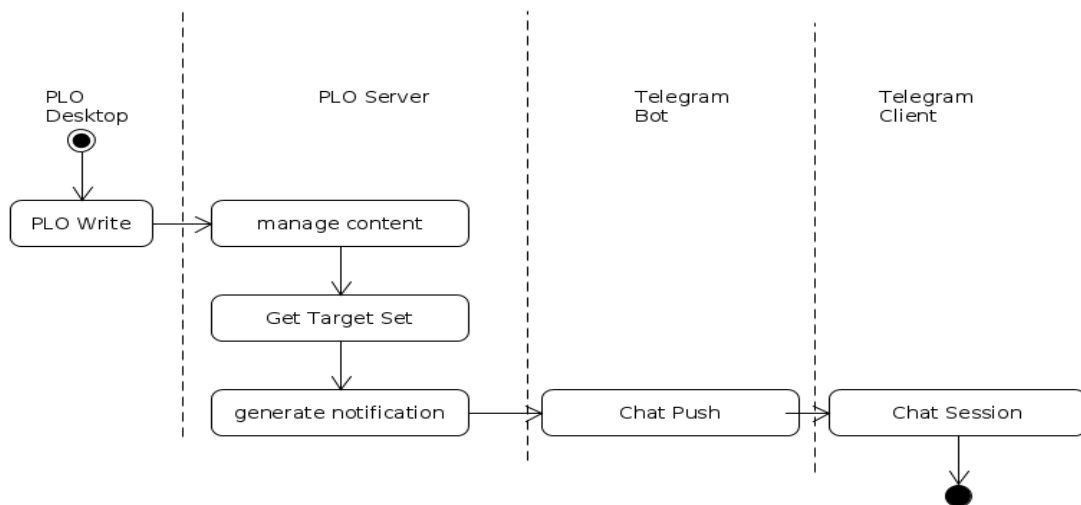


Figure 4 Messages in desktop communication entering the Telegram Chat

Sending PLO messages

PLO users must respond to the archive / memo messages they receive if the message is a task message. If the execution of the order is carried out in the near future, the response can be delayed until it is finished as well as a report. If the execution of the command requires a relatively long time, then the response will contain the message that it has been received and also the implementation plan. When done from the Telegram, the response must use the Reply mechanism so that the server can find out the corresponding message chain presented in Figure 5.

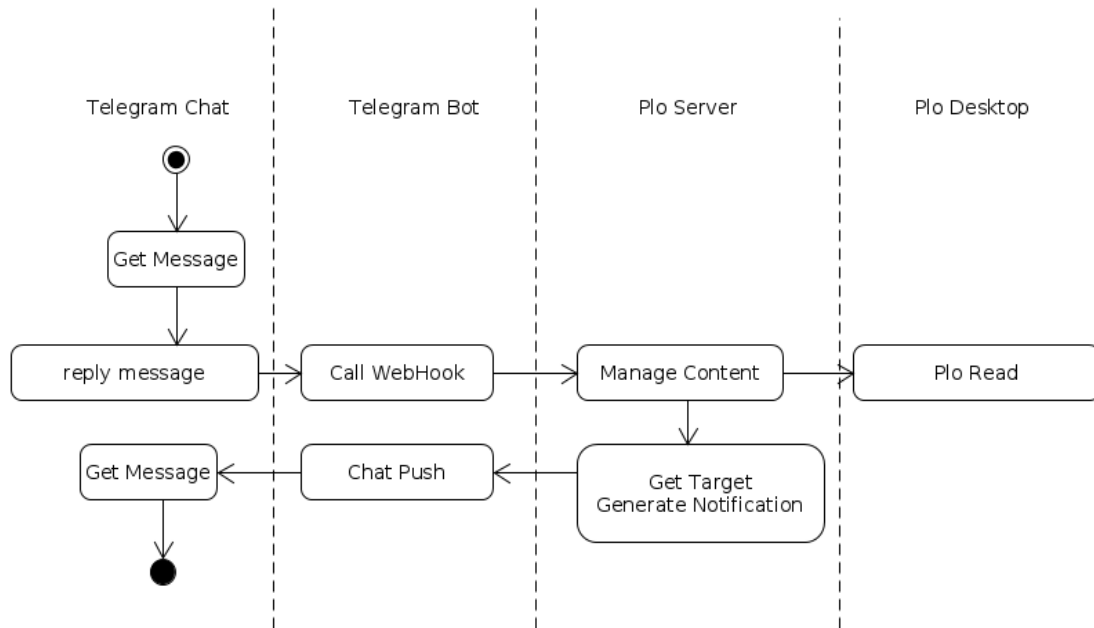


Figure 5 Mechanism for sending PLO messages to the telegrams chat

3.2 Implementation

Chat bots are implemented in the PHP programming language environment that is placed on the PLO server itself. The *Call* process to the telegram API embedded in the PLO module that contains the function of sending notifications. Chat messages are treated the same as the PLO notification system via email. The webhook telegram bot module carries out the *message write* functions.

3.2.1 Bot activation

Bot activation is implemented by storing activation codes and chat numbers in the multi-purpose attribute storage table. The PLO has provided a multipurpose table storing attributes for certain users with user identity fields, attribute names, numeric values and text values. The activation code is implemented as a round six digit number. When prompted, PLO saves with the *tcode* attribute name. For security, the string segment is filled in with text retrieval time. If within a certain time the code is not used, the PLO will delete it. The program flowchart snippet making the activation number with the PHP language is shown in Figure 6.

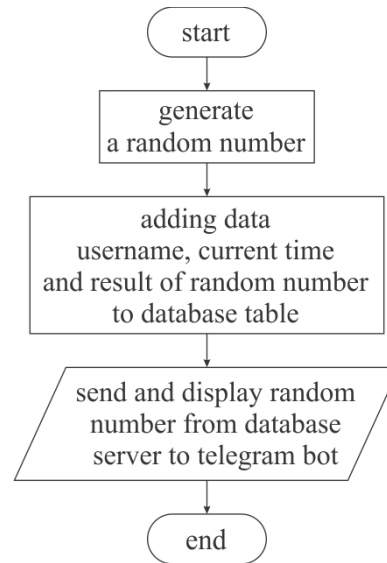


Figure 6 Flowchart of activation number program

Figure 6 shows a simple code to make a six digit random number given to the user as an activation code. When receiving the code, the telegram bot will record the session number and then use the *push* message.

3.2.2 Chat session

Telegram Bot connected with PLO is done using the relevant chat number. This chat number is stored with the name chatID attribute in a multipurpose storage area. The string segment can be filled with the connection date and the last usage date for audit purposes in the event of a security incident.

3.2.3 PLO messages

All archive / memo messages sent to each user are poured directly into the telegram bot chat session. This message's format starts with the word 'memo' in the corresponding with the chain number without being separated by a space or comment if it's a comment. Using this number, if the user sends a memo message, the system can find the PLO thread number in question.

3.3 Multi-threaded PLO

Basically PLO organizes multithreaded communication. Each parent message can be followed by a number of comments from the target reader of the parent message. It forms a communication thread which is formed with the user in the target set of readers as members. The telegram bot has limitations only providing one chatID for each user. Thus, the user who wants to send a message to a particular thread must use the reply mechanism to the message coming from the relevant thread.

3.4 Synchronous Messaging

The bot sends a complete PLO message to each target reader whenever there is a new message. The delivery is carried out sequentially with the sustainability of the processing response if the message has been sent to the target. Illustration of message delivery is shown in Figure 7.

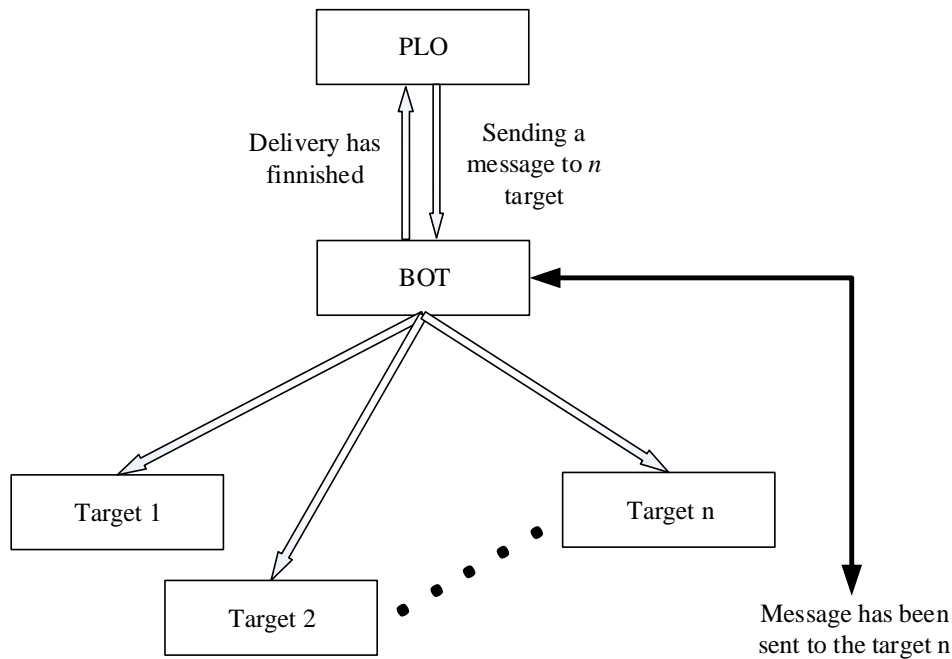


Figure 7 The message delivery from PLO to the targets

The average time for bots to send the messages is about 1.5 seconds per telegram account or per target. Therefore, the more messages targeted to the account, the more the operational delay will occur in the PLO desktop. In this study, delays began to be felt when the messages were targeted to more than 20 accounts.

4. CONCLUSION

The telegram bot can be used as an access interface to the PLO system. The PLO multithread can be accessed through the reply mechanism. The security for accessing the PLO via bot is guaranteed because of the unique chatID for each bot user. However, the larger number of messages sent by bots the larger delays will be happened, and thus will disrupt the operation of PLO desktops.

Subsequent research that can be done is the design and the implementation of image and file object viewers. In addition, it can also be designed and fully implemented in other PLO features such as *news*, *curah gagasan* dan *event calendar*. Also, it can be added an asynchronous delivery mechanism for messages via bots so it won't interfere the PLO desktop operation.

REFERENCES

- [1] R. Ling and C.H. Lai, "Microcoordination 2.0: Social Coordination in the Age of Smartphones and Messaging Apps", *Journal of Communication*, vol. 66, issue 5, pp. 834–856, Oct. 2016 [Online]. Available: <http://onlinelibrary.wiley.com/doi/10.1111/jcom.12251/full>. [Accessed: 4-Jan-2017]
- [2] H. T. Rowan-Kenyon, A. M. Martínez. Alemán, K. Gin, B. Blakeley, A. Gismondi, J. Lewis, A. McCready, D. Zepp and S. Knight, "Social Media in Higher Education:

- ASHE Higher Education Report”, *Social Education in Higher Education*, vol. 42, no. 5, pp. 7–128, Sept. 2016 [Online]. Available: <http://onlinelibrary.wiley.com/doi/10.1002/aehe.20103/abstract>. [Accessed: 5-Mar-2017]
- [3] N. Ahmad and P. Orton, “Smartphones Make IBM Smarter, But Not As Expected”, *Technology and Development* 64, pp. 46–50, 2010 [Online]. Available: https://www.columbia.edu/~na2189/files/T+D_Smarthphones_Make_IBM_Smarter.pdf. [Accessed: 26-Apr-2016]
- [4] D. Derks, D. Duin, M. Tims and A.B. Bakker, “Smartphone use and work-home interference: The moderating role of social norms and employee work engagement”. *Journal Occupation and Organization Psychology*, vol. 88, pp. 155–177, Aug. 2014 [Online]. Available: <http://onlinelibrary.wiley.com/wo11/doi/10.1111/joop.12083/full>. [Accessed: 28-Mei-2016]
- [5] A. Azfar, K.-K. R Choo and L. Liu, “An Android Communication App Forensic Taxonomy”. *Journal Forensic Sciece*, 2016, vol. 61, is. 5, pp. 1337–1350, Jul. 2016 [Online]. Available: <http://dx.doi.org/10.1111/1556-4029.13164>. [Accessed: 14-Feb-2017]
- [6] C. Montag, K. Błaszkiwicz, R. Sariyska, B. Lachmann, I. Andone, B. Trendafilov, M. Eibes and A. Markowetz, “Smartphone usage in the 21st century: who is active on WhatsApp?”, *BMC Resesarch Notes*, vol. 8, pp. 1–6, Aug. 2015 [Online]. Available : <https://bmcresnotes.biomedcentral.com/articles/10.1186/s13104-015-1280-z>. [Accessed: 6-Sept-2017]
- [7] T. T. Telegram, “100,000,000 Monthly Active Users”, 2016 [Online]. Available: <https://telegram.org/blog/100-million>. [Accessed: 28-Mar-2016]
- [8] T. T. Telegram, “Telegram Bot Platform”, 2015 [Online]. Available: <https://telegram.org/blog/bot-revolution>. [Accessed: 28-Mar-2016]
- [9] B.N. Prastowo, K. Mustofa, J. Istiyanto, and A. Ashari, “Towards Formal Specification of PaperLess Office Systems”, *Seminar Nasional Teknologi Informasi. Universitas Islam Indonesia*, Yogyakarta, 2013.
- [10] F. S. Amin and B.N. Prastowo, “Implementasi Application Programming Interface (API) pada Sistem Kontak UGM”, Universitas Gadjah Mada, 2015.
- [11] B. Travis, “Check out the new AI-powered TechCrunch news bot on Telegram messenger”, *TechCrunch TechCrunch-BLOG*, 2016 [Online]. Available: <http://search.proquest.com/docview/1773599597/A51C307B55B644E5PQ/1>. [Accessed: 28-Mar-2016]
- [12] J. Constine, “Assist Has No App. It’s A Free Travel/Shopping Chat Bot For SMS And Facebook Messenger”, *TechCrunch TechCrunch-BLOG*, 2016 [Online]. Available: <http://search.proquest.com/docview/1754422258/ACA7E55A1E564146PQ/3>. [Accessed: 28-Mar-2016]
- [13] Mena Report, “Sberbank launches virtual assistant for Telegram”, *MENA Rep*, Russian Federation, 2016.

-
- [14] T. Sutikno, L. Handayani, D. Stiawan, M.A. Riyadi and I. M. I. Subroto. “WhatsApp, Viber and Telegram: which is the Best for Instant Messaging?”, *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 6, no. 3, pp. 909-914, June 2016 [Online]. Available : <http://iaesjournal.com/online/index.php/IJECE/article/view/10271>. [Accessed : 24 June 2017]
- [15] A. Abashev, R. Grigoryev, K. Grigorian dkk. “Programming Tools for Messenger-Based Chatbot System Organization: Implication for Outpatient and Translational Medicines”, *Bio Nano Science – Springer*, vol.7, Issue 2, pp. 403-407, 2017 [Online]. Available : <https://doi.org/10.1007/s12668-016-0376-9>. [Accessed: 4-Oct-2017]
- [16] R. Sari-Motlagh, S. Ebrahimi, A. Nikfallah, S. Hajebrahimi, and B. Shakiba, “Lifelong Learning in Practice: The Age of Discussion through Social Media”, *European Urology*, vol. 69, Issue 6, pp. 1162-1163, June 2016[Online]. Available : [http://www.europeanurology.com/article/S0302-2838\(16\)00165-2/fulltext](http://www.europeanurology.com/article/S0302-2838(16)00165-2/fulltext). [Accessed : 4-Oct-2017]
- [17] J. Lockhart, 2015, *Modern PHP : New Features and Good Practices*, O’Reilly, Gravenstein Highway North, Sebastopol, USA, hal. 1-2.
- [18] A. R. Irawati and K. Mustofa, “Pengukuran Fungsionalitas Perangkat Lunak Menggunakan Metode Function Point Berdasarkan Dokumen Desain”, *Indonesian Journal of Computer Science (IJCCS)*, Vol. 7, No.2, pp. 111-120, 2013[Online]. Available: <https://jurnal.ugm.ac.id/ijccs/article/view/3348>. [Accessed : 12-Sept-2017]