

# Pengembangan Sistem Notifikasi pada LMS Moodle Melalui Penerapan Multi Channel Access Dengan Pemanfaatan Webhook

Rio Juniyantara Putra<sup>1</sup>, Dewa Made Wiharta<sup>2</sup>, Nyoman Putra Sastra<sup>3</sup>

[Submission: 12-04-2023, Accepted: 28-05-2023]

**Abstract**— The development of Moodle E-Learning technology has progressed rapidly to meet and support diverse learning needs. In the learning process of Moodle E-Learning, various activities are involved, such as creating assignments, providing assessments, and other activities. However, these processes have limitations, such as students not being immediately aware when assignments are created or when assessments are given. This often leads to delays in completing tasks as students wait for manual notifications from their instructors. Additionally, students sometimes forget to regularly check Moodle for updates. To address these shortcomings, the researcher aims to develop a MCA notification BOT application that integrates Moodle with social media platforms like Line and Telegram using webhooks. The purpose of this system is to help students receive real-time information about assignments and assessments provided on Moodle by their instructors. The testing results demonstrate that the implemented features function well and align with the intended design.

**Intisari**— Perkembangan teknologi E-Learning Moodle saat ini sudah berkembang pesat demi memenuhi dan mendukung kebutuhan proses pembelajaran yang beragam. Pada proses pembelajaran di *E-Learning Moodle* terdapat beberapa aktivitas seperti dosen membuat tugas, memberi penilaian, dan aktivitas lainnya. Semua proses di atas memiliki kendala seperti mahasiswa tidak mengetahui secara langsung saat dosen sudah membuat penugasan dan saat memberi penilaian terhadap tugas yang di berikan. Sehingga menyebabkan mahasiswa mengalami keterlambatan dalam pengerjaan tugas karna menunggu dosen memberikan informasi secara manual, dan terkadang mahasiswa lupa melakukan pengecekan moodle secara berkala. Berdasarkan kekurangan dan permasalahan di atas maka sebagai peneliti akan membangun aplikasi BOT notifikasi MCA sebagai media yang berintegrasi antara moodle dengan memanfaatkan *platform media sosial* seperti Line dan Telegram dengan memanfaatkan *webhook*. Tujuan dibangun sistem ini diharapkan dapat membantu mahasiswa untuk mendapatkan informasi penugasan, penilaian yang diberikan di *e-learning moodle* oleh dosen secara *real-time*. Hasil dari pengujian menunjukkan fitur yang bisa berfungsi dengan baik dan sesuai dengan apa yang di rancang.

<sup>1</sup>Mahasiswa, Program Pasca Sarjana, Manajemen Sistem Informasi dan Komputer Fakultas Teknik Universitas Udayana, Jln. P.B Sudirman Denpasar-Bali 80234 INDONESIA (telp: 082341469710; e-mail: [ryojuniyantara@gmail.com](mailto:ryojuniyantara@gmail.com))

<sup>2,3</sup>Dosen, Pascasarjana, Jurusan Teknik Elektro dan Komputer Fakultas Teknik Universitas Udayana, Jln. Jalan Kampus Bukit Jimbaran 80361 INDONESIA (telp: 0361-703315; fax: 0361-4321; e-mail: [wiharta@unud.ac.id](mailto:wiharta@unud.ac.id); [putra.sastra@unud.ac.id](mailto:putra.sastra@unud.ac.id))

**Kata Kunci**— Bot, LMS, MCA, Moodle, Webhook.

## I. PENDAHULUAN

Kemajuan teknologi informasi dan komunikasi sudah dimanfaatkan di berbagai segmen, sebagai contoh perusahaan dan instansi untuk mencapai kinerja yang maksimal. Lembaga pendidikan juga memanfaatkan teknologi untuk mencapai tujuannya. Salah satu bentuk implementasi teknologi di lembaga pendidikan yaitu *penggunaan Learning Management System (LMS) Moodle* sebagai *e-learning*. Pesatnya penggunaan LMS moodle di bidang pendidikan memberikan sisi positif terhadap aktivitas dan hasil pembelajaran [1]. Hal tersebut juga memengaruhi bertambahnya jumlah pengguna *e-learning moodle* di instansi pendidikan. Dengan adanya moodle, para pengajar atau dosen dapat dengan mudah membuat konten bahan ajar berbasis IT/e-learning [2].

Para pengajar atau dosen dalam pelaksanaan kegiatan *e-learning* di LMS moodle, kerap kali membuka moodle dan membuat informasi terkait penugasan, pemberian nilai, dan aktivitas lainnya. Namun terkadang mahasiswa tidak membuka moodle secara berkala yang menyebabkan mahasiswa tidak mendapatkan informasi mengenai jadwal-jadwal penting pengumpulan tugas dan quiz-quiz secara *real-time*, sehingga dapat menyebabkan keterlambatan ataupun terlewatkan pengumpulan tugas ataupun pengerjaan quiz.

Penelitian mengenai notifikasi secara *real-time* telah dilakukan oleh beberapa peneliti. Penelitian [3] dan [4] mengimplementasikan plugin notifikasi moodle sebagai media untuk integrasi antara *e-learning moodle* dengan Bot Telegram. Kedua penelitian ini hanya berfokus pada integrasi dengan Bot Telegram, sedangkan peserta atau mahasiswa tidak semua menggunakan aplikasi Telegram. Untuk itu perlu dikembangkan sebuah metode agar notifikasi secara *real-time* ini dapat diimplementasikan pada kanal-kanal komunikasi berbeda, sehingga peserta dengan pengguna kanal berbeda dapat menerima notifikasi tersebut.

Berdasarkan kendala di atas maka penelitian ini bertujuan untuk mengembangkan sistem notifikasi pada moodle yang akan berintegrasi dengan beberapa aplikasi perpesanan, contohnya Line dan Telegram dengan memanfaatkan bot pada masing-masing aplikasi tersebut. Dalam penerapannya, metode *webhook* yang sering disebut dengan *web callback* [5] akan digunakan sebagai media perantara untuk pengiriman notifikasi dari moodle ke Telegram dan Line Bot, sehingga nantinya pesan notifikasi tersebut akan diteruskan ke pengguna lainnya yang terhubung dengan Bot. Pengembangan sistem ini



diharapkan dapat membantu peserta atau mahasiswa dalam hal menerima informasi penugasan, penilaian, yang diberikan di *e-learning moodle*.

## II. TINJAUAN PUSTAKA

Sebuah plugin notifikasi yang digunakan sebagai media untuk mengintegrasikan LMS Moodle dengan Bot Telegram telah dilakukan pada penelitian [3]. Hal ini bertujuan untuk membantu guru mengirimkan pemberitahuan jika ada pengumuman yang diberikan guru kepada siswa dari LMS Moodle ke aplikasi telegram siswa. Akan tetapi pada penelitian ini hanya berfokus pada satu kanal, yaitu melalui aplikasi telegram.

Penelitian [6] merupakan LMS Moodle dengan model dan penggunaan yang mudah serta ringan penggunaannya, dan kelebihan ini dijadikan sebagai acuan untuk alasan penggunaan bot telegram. Penelitian ini memanfaatkan Bot Telegram untuk memandu siswa dalam melakukan pembelajaran secara online. Namun sekali lagi penelitian ini masih menggunakan satu kanal yaitu menggunakan Bot Telegram yang tersedia pada aplikasi telegram sebagai media perantara antara Moodle dengan siswanya.

Selanjutnya, penelitian [7] telah dibangun sebuah sistem informasi *e-learning* dan ujian yang dibuat berbasis file dengan memanfaatkan Bot Telegram sebagai media pendukung untuk *e-learning* yang dibangun. Dengan begitu mahasiswa dapat menjalankan perkuliahan dengan biaya yang lebih hemat. Sistem yang dibuat memungkinkan dosen dapat membuat soal ujian untuk sistem, sedangkan fitur mahasiswa dapat mengakses fitur di dalam *e-learning* untuk mengikuti serangkaian kegiatan perkuliahan dan mengirim tugas dengan cara mengupload file. Penelitian ini diimplementasikan dengan memanfaatkan satu kanal yaitu Telegram Bot saja dan belum bisa menyediakan dukungan untuk aplikasi lainnya.

Penelitian [8] membangun sebuah sistem yang bertujuan untuk dapat membuktikan bahwa media pembelajaran yang dibangun dengan menggunakan Bot Telegram dapat memberikan efektivitas yang dapat meningkatkan hasil belajar serta dapat memberikan suasana lingkungan pembelajaran yang efektif dan interaktif.

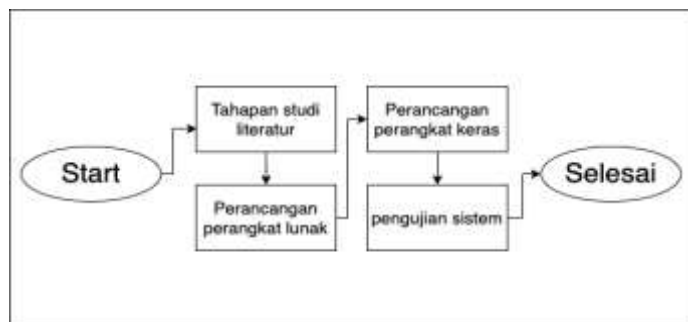
## III. METODE PENELITIAN

Tahapan penelitian berupa diagram alir ditunjukkan pada Gambar 1. Metode yang digunakan pada penelitian ini adalah eksperimental dengan tujuan untuk menguji sistem yang sudah dirancang. Secara umum diagram alir dan penjelasan dari penelitian ini adalah sebagai berikut.

### A. Study Literatur

Tahap penelitian ini dimulai dengan melakukan *study literatur*, perancangan perangkat lunak, perancangan perangkat keras, sampai dengan uji coba pada sistem. *Study literatur* pada penelitian ini dilakukan dengan cara mengumpulkan informasi pada artikel-artikel dari jurnal referensi yang dapat dijadikan acuan dasar dalam penyusunan laporan. Artikel yang digunakan sebagai referensi yaitu artikel yang berkaitan dengan

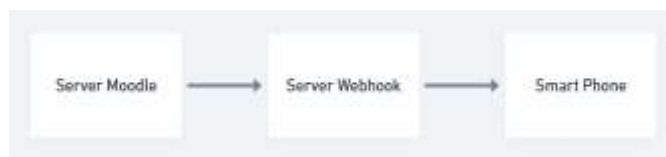
implementasi implementasi *e-learning* menggunakan LMS Moodle dengan pengembangan sesuai dengan kebutuhannya.



Gambar 1: Gambaran metode pengembangan waterfall.

### B. Rancang Bangun Perangkat Keras

Perangkat keras untuk implementasi Multi Channel Access (MCA) pada LMS Moodle menggunakan 1 perangkat komputer sebagai server Moodle yang terinstall LMS Moodle dan plugin webhook pada moodle. Selanjutnya dihubungkan dengan aplikasi lain yang bertugas menyediakan layanan webhook dan juga menyediakan layanan pengirim pesan ke Telegram Bot, dan Line Bot. Perangkat *smartphone* digunakan untuk menginstall aplikasi Telegram dan Line yang akan digunakan untuk uji coba. Hal ini ditunjukkan pada Gambar 2.



Gambar 2: Gambaran rancangan perangkat keras.

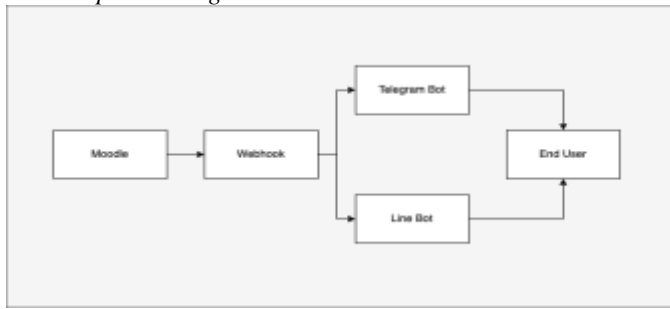
### C. Rancang Bangun Perangkat Lunak

Perancangan perangkat lunak untuk pengembangan MCA pada LMS Moodle dibagi menjadi 3 bagian, yaitu 1) Moodle yang terinstall plugin webhook untuk memonitor aktivitas di moodle, 2) webhook untuk menampung data aktivitas yang dikirim, dan 3) Bot Telegram dan Bot Line yang dikirim ke pengguna yang sudah menginstall Line dan Telegram yang terinstall di *Smart Phone*-nya. Hal ini ditunjukkan pada Gambar 3.

Pengembangan sistem ini didukung oleh beberapa tools atau aplikasi yang dapat dijelaskan sebagai berikut.

#### 1. Moodle

Moodle merupakan sebuah perangkat lunak yang berbasis *open Course Management System* yang didistribusikan secara gratis dan bebas dikembangkan [1][9][10][11][12]. Moodle dibangun dengan menggunakan bahasa pemrograman PHP oleh Martin Dougiamas dengan tujuan sebagai alat untuk disertainya [3].



Gambar 3: Gambaran rancangan perangkat lunak.

## 2. Bot

Bot merupakan sebuah hasil dari pemrograman komputer yang setiap prosesnya sudah ditentukan oleh pengembang Bot tersebut terlebih dahulu dan secara otomatis [13].

## 3. Webhook

Webhook merupakan konsep *Application Programming Interface (API)* yang biasa disebut *callback* [14]. Webhook terdiri dari link URL yang disiapkan agar data yang dikirim dapat langsung diterima pada waktu yang sama dengan link URL yang sudah disediakan [14]. Telegram menyediakan beberapa Bot Engine pada metode komunikasi, yaitu Long Polling dan Webhook[15].

## 4. Line

Line merupakan aplikasi perpesanan secara instan yang gratis, dan dapat digunakan di berbagai perangkat/platform, contohnya *smartphone*, tablet, dan komputer [16]. Line mengembangkan beberapa fitur baru yaitu *sticker* atau gambar yang dapat dikirim di ruang *chat* sebagai penggambaran ekspresi pengguna [17].

## 5. Line Bot

Line Messenger selain sebagai aplikasi perpesanan, line juga memiliki fitur Bot yang tersedia secara gratis, yang nantinya akan digunakan dan dihubungkan oleh pihak ketiga [18].

## 6. Telegram Bot

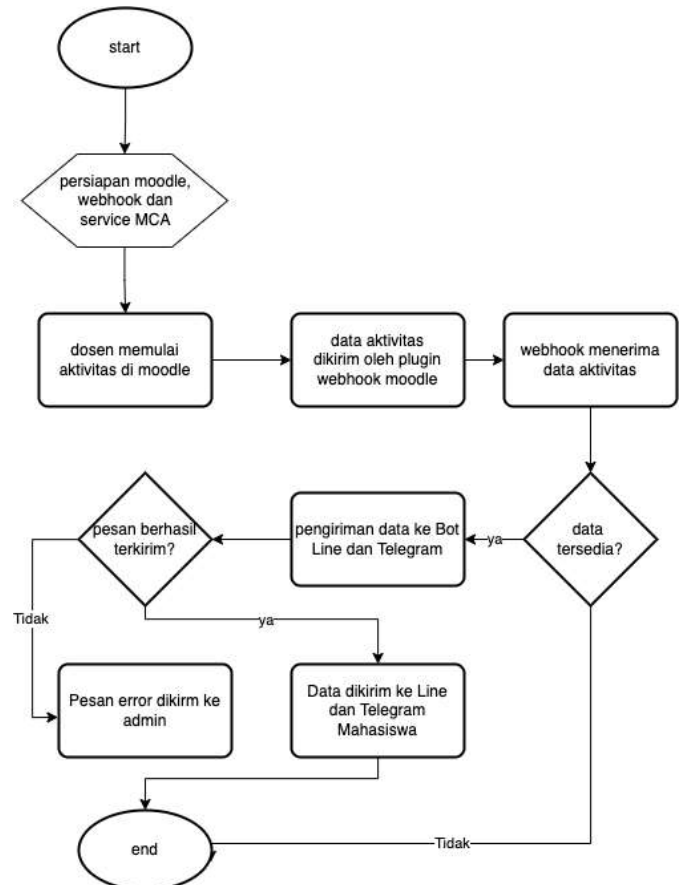
Telegram Bot merupakan fitur yang dijalankan pada aplikasi telegram, digunakan sebagai mitra dan memiliki fungsi tambahan lainnya, seperti melihat pesan, mengirim pesan, mengirim gambar dan lain-lain [19].

## 7. Javascript dan Nodejs

Javascript merupakan salah satu bahasa pemrograman, untuk membuat HTML menjadi lebih dinamis dengan memberikan perintah eksekusi pada sisi client [20]. Javascript memungkinkan pengembang untuk menggunakan fitur-fitur yang kompleks pada sebuah website [21]. Sedangkan NodeJS merupakan perangkat lunak yang pada dasarnya sudah dirancang untuk mengembangkan aplikasi web disisi server dan ditulis dengan javascript. NodeJs merupakan turunan dari

javascript yang memiliki sifat *non-blocking*, sehingga nodejs dapat menerima banyak request secara bersamaan [23]. NodeJS juga bisa disebut sebagai sebuah runtime environment [24]. Javascript dan NodeJS pada penelitian ini digunakan untuk membangun webhook [22].

Proses kerja atau tahapan dari sistem yang dikembangkan berupa *flowchart* dari alur sistem ini dapat dilihat pada Gambar 4.



Gambar 4: Flowchart sistem.

Berdasarkan *flowchart* yang dipaparkan pada Gambar 4, hal pertama yang dilakukan adalah memastikan moodle sudah siap digunakan dan terpasang plugin webhook yang sudah berisi konfigurasi alamat IP atau domain dari aplikasi webhook yang dibuat. Selanjutnya, jika ada aktivitas tertentu yang dilakukan oleh dosen, data akan dikirim ke aplikasi webhook yang sudah disediakan. Jika aplikasi webhook menerima data yang dikirim tersebut, webhook secara otomatis akan mengirim data tersebut ke Bot Telegram dan Line Bot. Dari Bot tersebut pesan yang berisi informasi dari data sebelumnya akan dikirim ke aplikasi Telegram dan Line mahasiswa yang terdaftar pada kuliah atau *course* tersebut.



D. Pengujian

Pengujian dilakukan dengan percobaan melakukan beberapa aktivitas di dalam moodle, sesuai Tabel I. pengiriman data aktivitas moodle akan dipastikan sampai ke Telegram dan Line pengguna yang tentu nantinya sekaligus memastikan data melewati service webhook dan pengiriman pesan ke Telegram Bot dan Line Bot sampai ke tahap akhir yaitu ke mahasiswa.

TABEL I  
DAFTAR PENGUJIAN

No	pengujian
1	Pengujian pengiriman pesan saat dosen membuat <i>assignment</i> pada <i>course</i> .
2	Pengujian pengiriman pesan saat dosen membuka <i>grading form</i> .
3	Pengujian pengiriman pesan saat dosen memberikan nilai pada halaman <i>grading form</i> .

IV. HASIL DAN PEMBAHASAN

Pembahasan dalam tahap ini mencakup implementasi dan pengujian sistem pada sistem yang telah dibangun dengan tujuan apakah sistem sudah berjalan dengan baik dan benar sesuai dengan prosedur dan rancangan yang telah dibuat. Dalam pengujian pada sistem ini akan dilakukan dengan menggunakan metode blackbox. Metode blackbox digunakan untuk melakukan pengujian yang difokuskan pada keluaran yang dihasilkan dari input yang diberikan. Pengujian akan dilakukan sampai ke tahap pengiriman pesan ke perangkat *smart phone* pengguna yang telah terinstall Line dan Telegram.

A. Pengujian pengiriman pesan saat dosen membuat tugas atau *assignment*

Tahap ini yang diuji adalah pengiriman pesan saat dosen membuka halaman *course* yang tersedia di moodle. Hal ini bertujuan untuk menguji apakan pesan dapat dikirim ke Telegram dan Line mahasiswa saat admin membuat tugas pada sebuah *course* dan memberikan informasi ke mahasiswa. Tahap pertama dilakukan adalah pembuatan *assignment* pada halaman *course* sudah tersedia. Hal ini dapat dilihat pada Gambar 5.



Gambar 5: Pembuatan assignment pada sebuah course

Setelah membuat *assignment* pada *course*, data akan terkirim ke Line dan Telegram seperti ditunjukkan pada Gambar 6 dan Gambar 7



Gambar 6: Pengiriman pesan pembuatan tugas dari moodle ke telegram

Gambar 6 menunjukkan penerimaan informasi yang dikirim oleh dosen di aplikasi moodle saat melakukan proses pembuatan tugas dan data tersebut ditampilkan ke dalam Telegram.



Gambar 7: Pengiriman pesan pembuatan tugas dari moodle ke line

Gambar 7 menunjukkan penerimaan informasi yang dikirim oleh dosen di aplikasi moodle saat melakukan proses pembuatan tugas dan data tersebut ditampilkan ke dalam aplikasi Line mahasiswa.

TABEL II  
 PENGUJIAN BLACKBOX PESAN SAAT DOSEN MEMBUAT ASSIGNMENT

Skenario	Output yang di harapkan	Output hasil
Pesan diterima saat dosen mahasiswa membuat assignment pada course di moodle	Chatbot memberikan informasi bahwa dosen sedang membuat assignment	Valid response seperti gambar 6, dan gambar 7

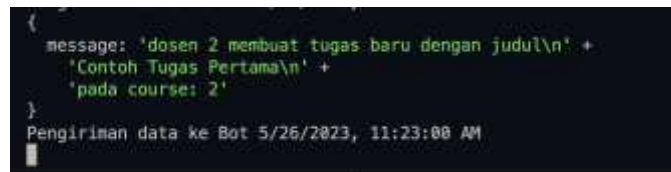
Berdasarkan hasil pengujian *blackbox* pada Tabel II dapat disimpulkan bahwa eksekusi sudah dijalankan dan hasil sudah sesuai dengan data yang dikirimkan dari aplikasi moodle dan yang didapatkan oleh mahasiswa.



Gambar 8: Hasil pengiriman data ke webhook

Gambar 8 menunjukkan pengujian pengiriman data dari Moodle ke aplikasi webhook yang sudah dibangun berjalan dengan baik dengan ditandai dengan *response code 200* dan durasi pengiriman data y 120.35 ms.

Rio Juniyantra Putra: Pengembangan Sistem Notifikasi Pada ...

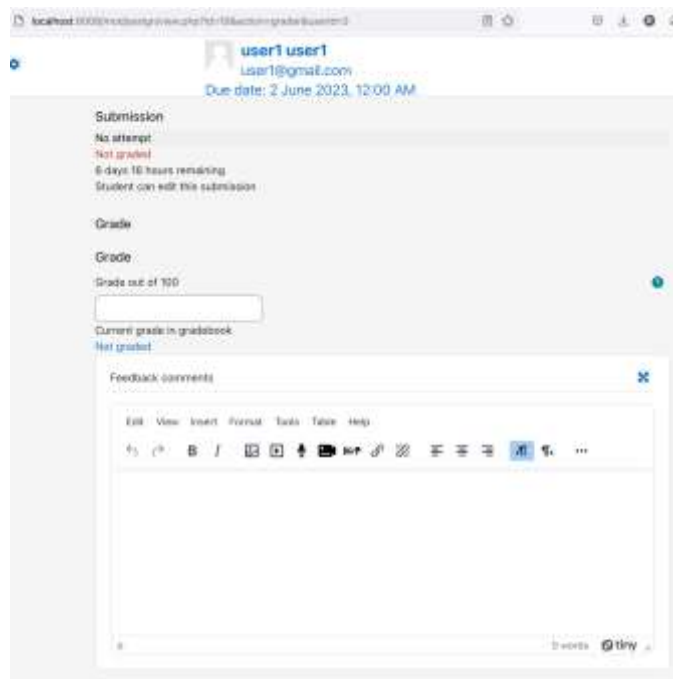


Gambar 9: Hasil pengiriman data dari webhook ke Bot

Gambar 9 menunjukkan hasil pengujian pengiriman data dari webhook ke aplikasi Telegram dan Line mahasiswa yang sudah dibangun berjalan dengan baik dengan ditandai dengan pesan yang dikirim sudah sesuai dan waktu pengiriman data yang sama.

### B. Pengujian pengiriman pesan saat dosen membuka *grading table* dan *grading form*

Pengujian selanjutnya yaitu pengujian pengiriman pesan saat dosen membuka *grading form* yang bertujuan untuk menampilkan form pemberian nilai terhadap tugas yang sudah diajukan oleh mahasiswa.



Gambar 10: membuka halaman grading form

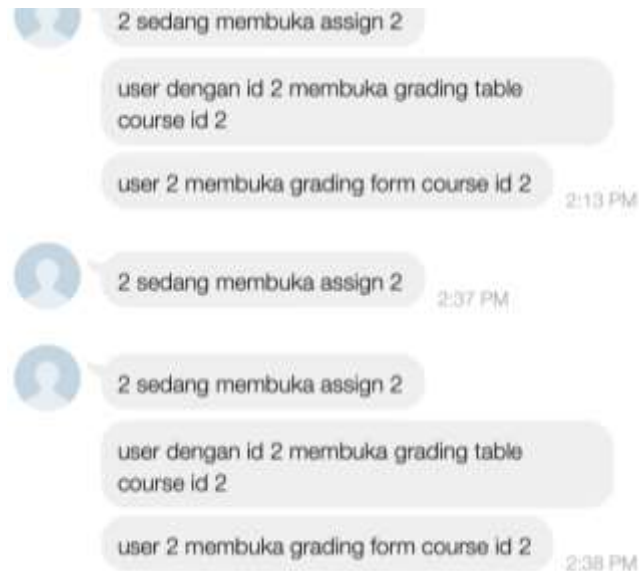
Gambar 10 merupakan halaman grading form, halaman ini merupakan halaman yang digunakan untuk memberi penilaian pada mahasiswa yang sudah mengumpulkan tugas yang disajikan sebelumnya oleh dosen.





Gambar 11: Pengiriman pesan informasi grading form dari Moodle ke Telegram

Gambar 11 menunjukkan penerimaan informasi yang dikirim oleh dosen dari aplikasi moodle melalui Bot Telegram saat melakukan proses membuka halaman *grading*, selanjutnya data tersebut ditampilkan ke dalam Telegram mahasiswa.



Gambar 12: Pengiriman pesan informasi grading form dari Moodle ke aplikasi Line

Gambar 12 menunjukkan penerimaan informasi yang dikirim oleh dosen dari aplikasi moodle melalui Bot Line saat

melakukan proses membuka halaman *grading*, selanjutnya data tersebut ditampilkan ke dalam aplikasi Line mahasiswa.

TABEL III  
PENGUJIAN PESAN SAAT MEMBUKA *GRADING FORM*

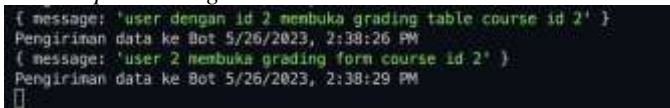
Skenario	Output yang di harapkan	Output hasil
Pesan diterima saat dosen membuka form nilai kepada mahasiswa di grading form	Chatbot memberikan respon berupa pesan <i>grading form</i> sudah di buka	Valid response seperti Gambar 11, dan Gambar 12

Berdasarkan hasil pengujian *blackbox* pada Table III dapat disimpulkan bahwa eksekusi pengujian pesan saat dosen membuka *grading form* sudah dijalankan dan hasil sudah sesuai dengan data yang dikirimkan dari aplikasi moodle dan yang didapatkan oleh mahasiswa.



Gambar 13: Hasil pengiriman data grading form ke webhook

Gambar 13 menunjukkan pengujian pengiriman data dari Moodle ke aplikasi webhook yang sudah dibangun berjalan dengan baik dengan ditandai dengan *response code 200* dengan durasi pengiriman data 1.54 ms.

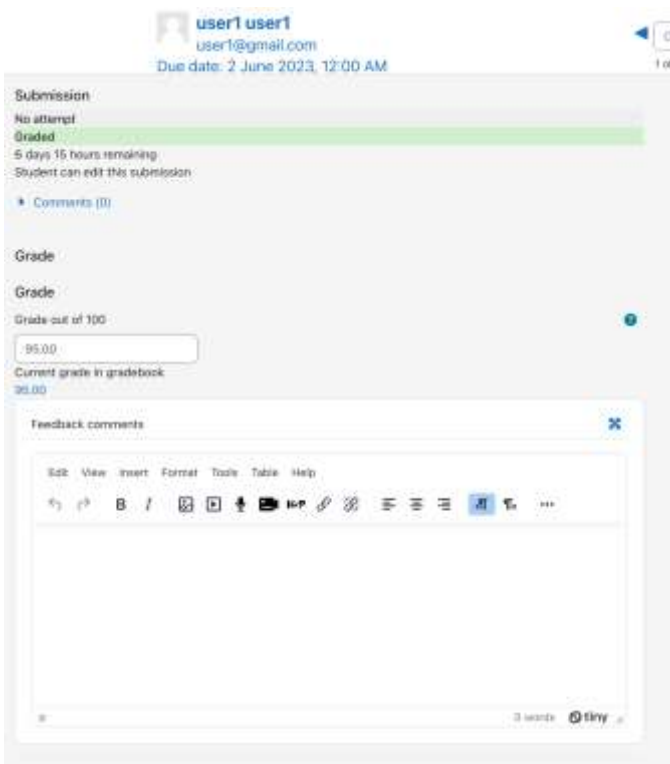


Gambar 14: Hasil pengiriman data *grading form* dari webhook ke Bot

Gambar 14 menunjukkan hasil pengujian pengiriman data berupa informasi *grading form* dari webhook ke aplikasi Telegram dan Line mahasiswa yang sudah dibangun berjalan dengan baik dengan ditandai dengan pesan yang dikirim sudah sesuai dan waktu pengiriman data yang sama.

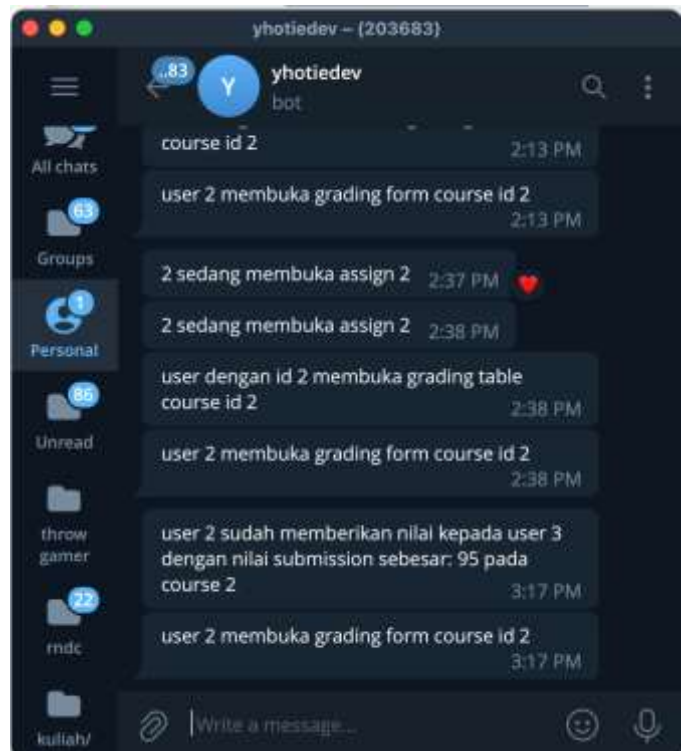
### C. Pengujian pengiriman pesan saat dosen memberikan nilai pada halaman *grading form*.

Pengujian selanjutnya yaitu pengujian pengiriman pesan saat dosen memberikan nilai pada tugas yang diajukan, dengan bertujuan untuk agar nilai tersebut dapat dilihat oleh pengguna di Bot Line maupun Bot Telegram. Hasil pengujian dan pembahasan akan diperlihatkan pada Gambar 15.



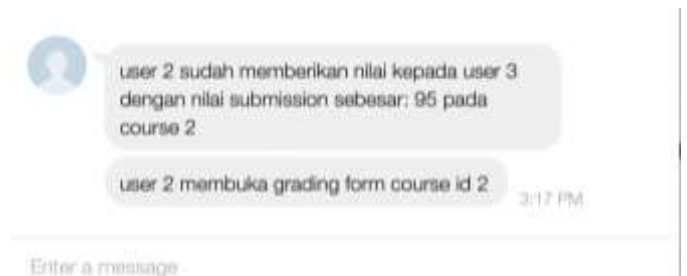
Gambar 15: membuka halaman *grading form*

Gambar 15 merupakan bagian yang digunakan untuk memberi nilai kepada mahasiswa. Grading dapat dilakukan pada form yang memiliki label *grade*. Di form inilah data nilai akan diinput dan menekan tombol *save change*. Maka nilai akan tersimpan di database dan data akan dikirim juga ke Telegram dan Line mahasiswa.



Gambar 16: Penerimaan pesan pemberian nilai oleh bot telegram dari Moodle

Gambar 16 menunjukkan penerimaan informasi yang dikirim oleh dosen dari aplikasi moodle melalui Bot Telegram saat dosen melakukan proses pemberian nilai pada halaman *grading*, selanjutnya data tersebut ditampilkan ke dalam Telegram mahasiswa melalui Bot Telegram.



Gambar 17: Penerimaan pesan pemberian nilai oleh bot line dari Moodle.

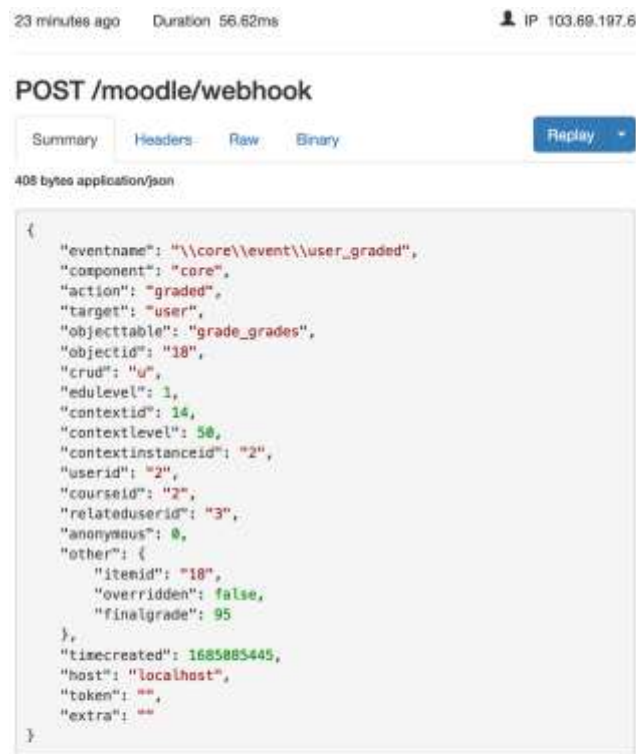
Gambar 17 menunjukkan penerimaan informasi yang dikirim oleh dosen dari aplikasi moodle melalui webhook dan melalui Bot Line saat dosen melakukan proses pemberian nilai pada halaman *grading*, selanjutnya data tersebut ditampilkan ke dalam Aplikasi Line mahasiswa melalui Bot line yang sudah disediakan.



TABEL IV  
PENGUJIAN PESAN SAAT MEMBUKA GRADING FORM DAN MEMBERI PENILAIAN

Skenario	Output yang di harapkan	Output hasil
Pesan diterima saat dosen memberikan nilai kepada mahasiswa di grading form	Chatbot memberikan respon berupa nilai dari tugas yang sudah di ajukan	Valid response seperti Gambar 16, dan Gambar 17.

Berdasarkan hasil pengujian *blackbox* pada Table IV dapat disimpulkan bahwa eksekusi pengujian pesan saat dosen mengirim nilai pada halaman *grading form* sudah dijalankan dan hasil sudah sesuai dengan data yang dikirimkan dari aplikasi Moodle dan yang didapatkan oleh mahasiswa.



Gambar 18: Hasil pengiriman data penilaian dari moodle ke webhook

Gambar 18 menunjukkan pengujian pengiriman data penilaian tugas yang dikirim dari Moodle ke aplikasi webhook yang sudah dibangun dan sudah berjalan dengan baik dengan

ditandai dengan *response code 200* dengan durasi pengiriman data 56.62 ms.



Gambar 15: Hasil pengiriman data penilaian dari webhook ke Bot.

Gambar 15 menunjukkan hasil pengujian pengiriman data berupa informasi nilai dari tugas yang berasal dari webhook ke aplikasi Telegram dan Line mahasiswa yang sudah di bangun berjalan dengan baik dengan ditandai dengan pesan yang dikirim sudah sesuai dan waktu pengiriman data yang sama.

TABEL V  
PENGUJIAN DURASI PENGIRIMAN DATA DAN RESPONSE TIME

pengujian	webhook(milli second)	Pengiriman ke bot(second)
pengujian pengiriman pesan saat membuat <i>assignment</i>	120.35	1 s
pengujian pengiriman pesan saat membuka <i>Grading Form</i>	1.54 ms	1 s
pengujian pengiriman pesan saat memberikan penilaian	56.62 ms.	1 s

Berdasarkan Table V, uji coba pengujian kecepatan pengiriman data dan response time yang didapatkan, hasil pengujiannya bahwa komunikasi end-to-end yang diperoleh adalah *real-time*. Hasil yang diperoleh adalah pengiriman data dari Moodle ke webhook dengan durasi rata-rata 59.50 ms dan pengiriman rata-rata data dari webhook ke Bot Telegram dan Line adalah 1 detik.

V. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan dan pembahasan yang sudah dijabarkan, pada penelitian ini telah berhasil dibuat sebuah sistem yang dapat mengirim pesan beberapa aktivitas di moodle ke dua buah *platform messaging*, yaitu Line dan Telegram melalui Bot. Model pengiriman pesan pada beberapa platform ini merupakan kebaruan dalam penelitian ini. Chatbot yang digunakan untuk pengiriman pesan sudah dapat berkomunikasi dengan baik menggunakan platform *messaging* gratis yaitu Line dan Telegram.

Sistem webhook yang dibangun sudah bisa berintegrasi dengan plugin webhook yang sudah terinstall di moodle, hal ini ditunjukkan dengan berhasilnya data aktivitas diujicobakan dan berhasil dikirim dari moodle dan sudah dapat ditampilkan di aplikasi Line dan Telegram pengguna secara *real-time* dengan



durasi rata-rata pengiriman pesan ke webhook yaitu sebesar 59.50 mili detik. Begitu juga dengan durasi pengiriman data ke Bot Telegram dan Bot Line, durasi rata-rata yang dibutuhkan untuk pengiriman yaitu 1 detik agar pesan tersebut sampai ke mahasiswa.

Pengembangan model ini untuk ke depannya dapat dikembangkan agar mencapai sistem yang lebih optimal dengan implementasi menjalankan sistem di server *cloud* sehingga dapat dimanfaatkan oleh banyak pengguna. Layanan perpesanan yang terintegrasi adalah 2 layanan, yaitu Line dan Telegram dan nantinya diharapkan dapat diintegrasikan dengan layanan perpesanan lain, seperti pengiriman ke Email, Slack Messaging, Firebase Push Notification, dan Discord. Aktivitas moodle yang ditangani masih terbatas, diharapkan nantinya mampu melayani aktivitas penting lainnya sehingga pengguna dapat memantau ragam aktivitas di dalam moodle lebih banyak.

#### REFERENSI

- [1] D. H. SIMBOLON, "Implementasi Lms (Learning Management System) Moodle Terhadap Hasil Belajar Mahasiswa Di Universitas Quality," *J. Curere*, vol. 5, no. 1, p. 1, 2021, doi: 10.36764/jc.v5i1.549.
- [2] H. R. Fajrin, S. Widadi, and A. B. Raharja, "Implementasi Lms Moodle Untuk E-Learning Sd Muhammadiyah Maesan Dan Sd Muhammadiyah Bangeran, Yogyakarta," *Martabe J. Pengabd. Kpd. Masy.*, vol. 4, no. 1, p. 78, 2021, doi: 10.31604/jpm.v4i1.78-84.
- [3] R. Muhammad, I. Arwani, and B. Rahayudi, "Implementasi Plugin Notifikasi Sebagai Media Integrasi Antara E-Learning Moodle dengan BOT Telegram (Studi Kasus: Bimbingan Belajar The Second School)," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 4, no. 7, pp. 2275–2282, 2020.
- [4] A. Widiyono, "Pengaruh Penggunaan LMS dan Aplikasi Telegram terhadap Aktivitas Belajar," *J. Penelit. Ilmu Pendidik.*, vol. 14, no. 1, pp. 91–101, 2021, doi: 10.21831/jpipfip.v14i1.37857.
- [5] B. Z. Ferdian and E. S. Nugroho, "Sistem Informasi Rekapitulasi Pemilukada Kota Pekanbaru menggunakan Input dari Telegram API," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 1, no. 3, pp. 56–63, 2017.
- [6] M. Faid, T. Rahmati, and Z. Arifin, "Desain bot telegram menggunakan python 3.8 dalam membantu proses belajar dan mengajar secara daring," *NJCA (Nusantara J. Comput. Its Appl.)*, vol. 5, no. 2, pp. 79–84, 2020, [Online]. Available: <https://njca.co.id/main/index.php/njca/article/view/194>.
- [7] R. Parlika, H. Khariono, H. A. Kusuma, and A. Setyawan, "JIP (Jurnal Informatika Polinema) PEMANFAATAN BOT TELEGRAM SEBAGAI E-LEARNING UJIAN BERBASIS FILE," *JIP (Jurnal Inform. Polinema)*, vol. 7, no. 4, pp. 65–72, 2021, [Online]. Available: <http://jip.polinema.ac.id/ojs3/index.php/jip/article/view/696>.
- [8] N. R. Putra and D. A. Dermawan, "Efektivitas Media Pembelajaran Interaktif Menggunakan Bot Telegram Pada Kelas X Multimedia Pada Pelajaran Komputer Dan Jaringan," *J. IT-EDU.*, vol. 6, no. 2, pp. 69–76, 2021, [Online]. Available: <https://ejournal.unesa.ac.id/index.php/it-edu/article/view/43731/37309>.
- [9] M. H. Fahmi and B. S. I. Cipta, "Pengembangan Blended Learning Berbasis Moodle (Studi Kasus Di Universitas Islam Raden Rahmat Malang)," *J. Teknol. Terap. G-Tech*, vol. 2, no. 1, pp. 106–113, 2020, doi: 10.33379/gtech.v2i1.328.
- [10] L. Herayanti, S. Gummah, B. A. Sukroyanti, G. Gunawan, and M. Makhrus, "Pengembangan Perangkat Pembelajaran Berbasis Masalah Menggunakan Media Moodle Untuk Meningkatkan Keterampilan Berpikir Kritis Mahasiswa Pada Materi Gelombang,"

- [11] J. Pendidik. Fis. dan Teknol., vol. 4, no. 2, p. 158, 2018, doi: 10.29303/jpft.v4i2.803.
- [12] W. Herbimo, "Penerapan Aplikasi Moodle Sebagai Salah Satu Model Pembelajaran Jarak Jauh Di Masa Pandemi," *Ideguru J. Karya Ilm. Guru*, vol. 5, no. 1, pp. 107–113, 2020, [Online]. Available: [file:///C:/Users/ilham/Downloads/144-Article Text-467-1-10-20200621.pdf](file:///C:/Users/ilham/Downloads/144-Article%20Text-467-1-10-20200621.pdf).
- [13] F. Inggriyani, N. Fazriyah, and A. Purbasari, "Penggunaan E-learning Berbasis Moodle bagi KKG Sekolah Dasar di Kecamatan Lengkong Kota Bandung," *J. SOLMA*, vol. 8, no. 2, p. 268, 2019, doi: 10.29405/solma.v8i2.3695.
- [14] I. Penelitian and A. D. Mulyanto, "Pemanfaatan Bot Telegram Untuk Media," vol. 12, no. 1, pp. 49–54, 2020.
- [15] T. Istiana, R. Indra A, G. S. Budhi Dharmawan, and B. Prakoso, "Pengembangan Sistem Diseminasi Prakiraan Cuaca Menggunakan Aplikasi Bot Telegram dengan Metode Webhook," *Elektron J. Ilm.*, vol. 12, no. 1, pp. 41–47, 2020, doi: 10.30630/eji.12.1.159.
- [16] M. A. Aris Widya and P. Airlangga, "Pengembangan Telegram Bot Engine Menggunakan Metode Webhook Dalam Rangka Peningkatan Waktu Layanan E-Government," *Saintekbu*, vol. 12, no. 2, pp. 13–22, 2020, doi: 10.32764/saintekbu.v12i2.884.
- [17] M. S. s Kadaton and R. Soekarta, "Perancangan Sistem Informasi Pariwisata Kota Sorong Menggunakan Web Dan Line@ Chatbot Sebagai Media Informasi Pariwisata," *Insect (Informatics Secur. J. Tek. Inform.)*, vol. 4, no. 2, p. 71, 2019, doi: 10.33506/insect.v4i2.561.
- [18] M. Nasyaya and I. Adila, "Diversifikasi Fitur dan Kolonisasi Data pada LINE Social Messaging Features Diversification and Data Colonialism on LINE Social Messaging," *J. Kominfo*, vol. 8, no. 2, pp. 96–101, 2019, doi: 10.31504/komunika.v8i2.2459.
- [19] K. Darmaastawan, P. Lanang Bagus Suputra Jaya Amertha, and L. Jasa, "Sistem Pakar Diagnosis Penyakit Gigi dengan Metode Breadth First Search berbasis Instant Messaging LINE Messenger," *Maj. Ilm. Teknol. Elektro*, vol. 20, no. 1, p. 139, 2021, doi: 10.24843/mite.2021.v20i01.p16.
- [20] C. G. I. Raditya, P. A. S. Dharma, I. K. A. A. Putra, I. B. K. Sugirianta, and I. B. I. Purnama, "Pendeteksi Kebocoran Gas dan Kebakaran Dini Menggunakan NodeMCU Berbasis Telegram," *Maj. Ilm. Teknol. Elektro*, vol. 21, no. 1, p. 13, 2022, doi: 10.24843/mite.2022.v21i01.p03.
- [21] M. Fernando, L. Jasa, and R. S. Hartati, "Monitoring System Kecepatan dan Arah Angin Berbasis Internet of Things (IoT) Menggunakan Raspberry Pi 3," *Maj. Ilm. Teknol. Elektro*, vol. 21, no. 1, p. 135, 2022, doi: 10.24843/mite.2022.v21i01.p18.
- [22] R. Salim, D. Arisandi, and J. Hendryli, "Pembuatan Aplikasi MOSTRANS Transporter Berbasis Mobile Menggunakan React-Native JavaScript," *J. Ilmu Komput. dan Sist. Inf.*, vol. 10, no. 1, 2022, doi: 10.24912/jiksi.v10i1.17856.
- [23] I. Kurniawan, Humaira, and F. Rozi, "REST API Menggunakan NodeJS pada Aplikasi Transaksi Jasa Elektronik Berbasis Android," *JITSI J. Ilm. Teknol. Sist. Inf.*, vol. 1, no. 4, pp. 127–132, 2020, doi: 10.30630/jitsi.1.4.18.
- [24] S. Sauda and M. Barokah, "Penerapan Nodejs Dan Postgresql Sebagai Backend Pada Aplikasi Ecommerce Localla," *INFOTECH J.*, pp. 101–105, 2022.
- [25] R. FAJRIN, "Pengembangan Sistem Informasi Geografis Berbasis Node.JS untuk Pemetaan Mesin dan Tracking Engineer dengan Pemanfaatan Geolocation pada PT IBM Indonesia," *J. Inform.*, vol. 11, no. 2, pp. 40–47, 2017, doi: 10.26555/jifo.v11i2.a6090.



{Halaman ini sengaja dikosongkan}