Utilization of REST API in calculating the duration of participation of moodle LMS participants

Akhmad Zaini, Heri Santoso*, Ahmad Toyib Ramadhani

Universitas PGRI Kanjuruhan Malang, Indonesia
*Corresponding author, email: heri@unikama.ac.id

Article history
Submitted: 10 November 2025
Revised: 20 December 2025
Accepted: 20 January 2025
Published: 26 January 2025

Keywords
Branch and bound
Distribution routes
Nearest neighbour
Optimization

Abstract
Moodle is an open-source Learning Management System that is widely used by various educational institutions. Moodle supports different types of learning activities, both synchronous and asynchronous. As an established LMS, Moodle has been equipped with an API with various reporting facilities, both for analysis purposes and tracking participant participation. Participant participation can be known through activity log reports. In certain situations, the activity log report cannot meet the needs of the user, especially when a teacher wants to know how much time the participants spend in a class. This research was conducted in order to stick to the API activity log report generated by Moodle through recording the duration of participation. The process of recording the duration of participant participation is carried out in real-time through the Moodle plugin which will send activity data to the Log aggregator through the REST API mechanism. The use of REST APIs can reduce the workload of Moodle servers, as Log aggregators can be embedded on other servers outside of Moodle's servers.

1. Introduction

Moodle is an open-source e-learning platform that is widely used by various educational institutions, both at the school and university levels [1]. Moodle supports various types of learning activities, both synchronous and asynchronous [2]. This diversity of activities and free factors is what makes Moodle the right choice to apply to various types of educational institutions. The various learning communication functions that include discussion, reporting, progress tracking, organization and collaboration can be replaced by Moodle [3].

Although Moodle has provided a participant activity tracking feature, in certain situations this feature is not enough to meet the needs of its users [4]. Moodle itself has been attached to the API of a log feature that can be used to see a series of activities carried out by participants [5]. The log information provided by Moodle's activity logs includes: time, username, affected user, event context, component, event name, description, origin and IP address. If we look at all the standard log attributes generated, a teacher will find it difficult to know how long (duration) the participants participated in the Moodle system. One of the attributes that has the potential to be used as a basis in measuring the duration of participant participation is the time attribute. This attribute contains information about the date and time that participants perform an activity, but does not explain how long the participant spends performing an activity.

The calculation of the duration of participant participation can be done by calculating the difference between user sessions on the Moodle log table [6]. Calculating based on the difference between sessions requires us to set a difference threshold. The determination of the difference threshold aims to ensure that there is no recognition of duration on the difference beyond the specified threshold. The difference between sessions that exceeds the predetermined threshold can be caused by the user having been inactive in the browser for a long time, and then they want to continue their activities. Setting the difference threshold between sessions will be invalid for activities that have varying time completion dynamics. Moodle provides reading activities in the form of pages [7]. These reading activities have a very varied completion time. A participant can spend hours just reading 1 page activity, because their reading material is indeed very much.

Another alternative to calculating the duration of participation is to execute AJAX Requests periodically as long as the page is still open [8]. The AJAX request is sent to the server and then the server will make a change to the participation time interval. Sending an AJAX Request like this has a loophole if it is not accompanied by user session verification. Participants can end their browser session, but they still send Requests through tools such as postman or other exploiter tools.

1.1. AJAX (Asynchronous Javascript and Xml)

AJAX stands for Asynchronous Javascript and Xml. AJAX is the use of the XMLHttpRequest object to communicate with the server. AJAX allows us to send and receive various information in JSON (Javascript Object

© 2026 The Author(s)
This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License
ISSN: 3110-1577

Notation) format, as well as XML (eXtensible Markup Language), HTML and plain text [9]. Requests made by AJAX are made asynchronously and in realtime [10], which is why AJAX Requests can be executed without refreshing the page.

The use of AJAX to send participation data is the right choice, as users do not need to refresh the page they are visiting [9]. Nonetheless, the utilization of AJAX should also be accompanied by verification of user sessions, to ensure that the AJAX executor is actually from a legitimate client. The standard structure of the Moodle plugin has provided global variables that we can leverage to verify sessions through $SESSION and $USER variables [11].

1.2. REST ful API

REST (Representational State Transfer) is an architectural model in building hypermedia-based systems [12]. The REST architecture is essentially not tied to any protocol, but generally REST makes use of HTTP as its communication protocol. The key to REST architecture is statelessness. Data communication in a REST architecture runs stateless, which means that each communication request made runs independently of all previous requests.

API (Application Programming Interface) is an interface provided by an application system in order to communicate with the environment outside the program [13]. APIs allow system development to be more flexible because it is not tied to a specific platform or ecosystem. The current development of APIs has largely implemented REST architectures [14]. This combination of these two elements is often referred to as a REST ful API.

2. Method

Each activity in Moodle is represented by a special page, meaning that every time a participant performs an activity, it can be viewed through the page they open. The calculation of the duration of participation is done by calculating the length of time spent by the user to open a page. An important part of calculating the length of time spent by a user in opening a page is the Agent. An Agent periodically sends update trigger information to the Moodle server that a page is actively open. The agent will stop sending update triggers when the browser is closed. Moodle's servers will always send activity information to the Log aggregator whenever it gets a trigger from the Agent. Before sending user activity information, Moodle's servers will first confirm that the trigger sent by the Agent is actually from a legitimate Agent by verifying the token. The token will always change every time a user switches pages. This token verification is designed to avoid triggers being sent by fake Agents. Without token verification, the cheating user could continue to send triggers as Agents without opening the actual Moodle page. Activity chart recording duration of participation can be seen in Figure 1.

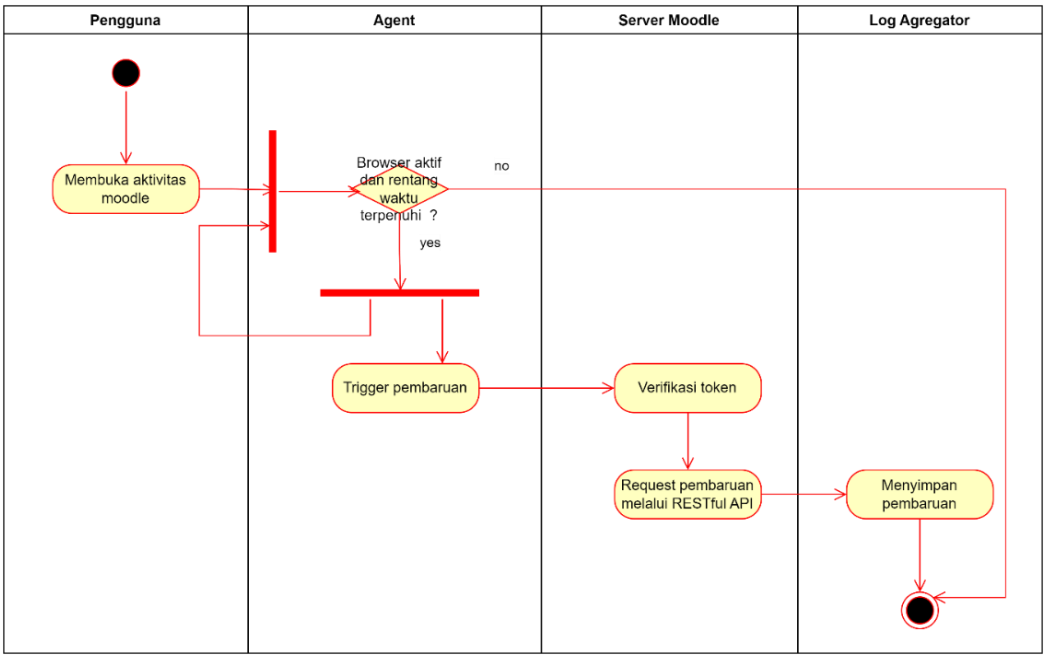

Figure 1. Activity Chart Recording Duration of Participation

The log aggregator is placed separately from the Moodle server. This is done to reduce the workload of the Moodle server, considering that updates of user activity information will always occur periodically as long as the user's page is active. Moodle servers only need to send update information without the need to do any process. Details of the process activities of each component can be seen in Figure 1.

The series of activities of the process of recording the duration of participation must be able to produce accurate information. The quality of Duration Logging success was measured by testing several aspects including token verification, Request REST ful API and duration logging accuracy.

3. Results and Discussion

3.1. Token Verification

The signal sent by the Agent must be ensured to be genuine from a legitimate Agent. The Agent implementation is packaged in a javascript format that is embedded in the client browser. Embedding the Agent in the client's browser has the potential to cause the sending of signals by fake Agents. Cheating users can send trigger signals that resemble Agents, so that the duration of participation can increase without having to open the actual Moodle page. To guarantee that the trigger signal sent by the Agent is really from the browser that is indeed active in opening the Moodle page, a verification process is required. Verification is packaged in the form of tokens, tokens are designed to always change every time the user turns the page. The agent that sends the trigger signal must include its token information. If a fake agent sends a trigger signal without including a token, then the server will send a 403 reply code, the failed attempt to send the Agent signal using the postman application can be seen in Figure 2.

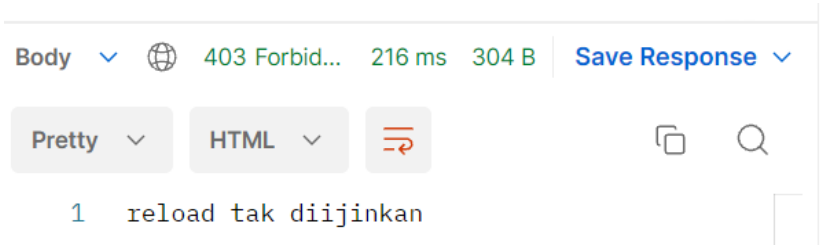

Figure 2. Verifikasi token dengan kode balasan 403

Token verification not only ensures the existence of the token, but the token must also be verified to be valid and comes from the browser that is opening the Moodle page. If a user deliberately sends a trigger signal without opening the Moodle page, the Moodle server will provide a 401 reply code. The failure to send a trigger signal by the Agent who imposes the Request without opening the Moodle page can be seen in Figure 3.

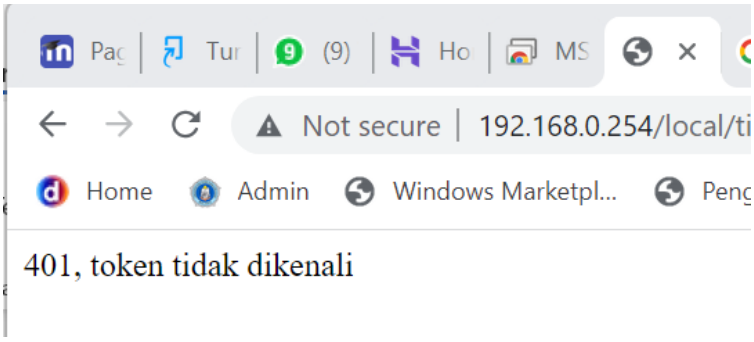

Figure 3. Verify the token with a 403 reply code

3.2. Request REST ful API

The Agent component of the participation duration recording system acts as a beacon. Continuously, the Agent sends a signal to the Moodle server to notify that the user is actively visiting the Moodle page. The trigger signal receiver from the Agent is in the form of a Moodle plugin which is a local plugin. The selection of a local plugin is carried out to make it easier for the Moodle server to communicate with an external system, namely the Log aggregator [15], through the REST ful API. The format of the API Request sent to the Log aggregator is as follows:

```
POST /index.php/timespend HTTP/1.1
Host: 192.168.0.253
Content-Length: 626
Content-Type: multipart/form-data; boundary=----
WebKitFormBoundary7MA4YWxkTrZu0gW

----WebKitFormBoundary7MA4YWxkTrZu0gW
Content-Disposition: form-data; name="userid"

1
----WebKitFormBoundary7MA4YWxkTrZu0gW
Content-Disposition: form-data; name="username"

Akhmad Zaini
----WebKitFormBoundary7MA4YWxkTrZu0gW
Content-Disposition: form-data; name="activity"

forum diskusi 1
----WebKitFormBoundary7MA4YWxkTrZu0gW
Content-Disposition: form-data; name="activity_type"

forum
----WebKitFormBoundary7MA4YWxkTrZu0gW
Content-Disposition: form-data; name="start"

2022-09-24 09:43
----
WebKitFormBoundary7MA4YWxkTrZu0gW
Content-Disposition: form-data; name="end"

2022-09-24 14:43
----WebKitFormBoundary7MA4YWxkTrZu0gW
Content-Disposition: form-data; name="url"

http://192.168.0.254/mod/page /view.php?id=289
----WebKitFormBoundary7MA4YWxkTrZu0gW
Content-Disposition: form-data; name="title"

tes_course: Page  189
----WebKitFormBoundary7MA4YWxkTrZu0gW
```

There are 7 pieces of information sent to the Log aggregator, namely userid, username, activity, activity_type, start, end, url and title. When a Request is made, the Log aggregator will respond:

```
{
    "respon": "ok with insert"
}
```

In the Log aggregator, security is no longer needed as in the case of the Agent, because we can secure the REST Request of the API to the Log aggregator by creating a separate network, where the network is only used for communication purposes between the Moodle server and the Log aggregator only.

3.3. Duration Recording Accuracy

Measurement of the recording duration is carried out by comparing the actual duration with the duration stored in the log aggregator database. We use the help of the TimeYourWeb Time Tracker plugin on the chrome browser to find out how long a page is open by a user [16]. The duration of the Log aggregator is calculated by performing an aggregation query, where each row in the selected data is multiplied by the signal transmission interval (5 seconds). A comparison of the actual duration with the duration in the Log aggregator can be seen in Table 1.

Table 1 shows the time difference between the actual duration and the duration in the Log aggregator database. Time differences can occur because the Agent does not always send a trigger signal. The agent sends a trigger signal when the timer has reached a certain interval. In the experimental configuration, we used an interval of 5 seconds. The determination of this interval aims to reduce the workload of the Log aggregator and

improve the efficiency of the use of storage capacity in the Log aggregator database. Although there is a time difference, it can be ignored, considering that the time difference that occurs never exceeds the predetermined interval (time difference <= 5 seconds).

Table 1. Comparison of Actual Duration with Log Aggregator Duration

No	Page title	Actual Duration	Duration of the Log Aggregator	Time Difference
1	Tes_course: Assignment 36	5 m, 33 d	5 m, 35 d	2 d
2	Tes_course: Assignment 13	4 m, 59 d	4 m, 55 d	1 d
3	Forum	4 m, 51 d	4 m, 50 d	1 d
4	Tes_course: Page 108	3 m, 56 d	3 m, 55 d	1 d
5	Tes_course: Assignment 80	3 m, 25 d	3 m, 25 d	0 d
6	Tes_course: Page 189	2 m, 30 d	2 m, 35 d	5 d
7	Tes_course: Page 134	1 m, 22 d	1 m, 20 d	2 d
8	Research Elearning: Log in to the site	13 d	10 d	3 d
9	Course: Kursus Ujicoba	7 d	5 d	2 d

4. Conclusion

Based on a series of experiments that have been conducted, it can be concluded that the system for recording the duration of participation can work according to expectations, both in terms of security (token verification), communication effectiveness (REST ful API), and duration accuracy. Another advantage obtained through the REST ful API is the efficiency on the server side of Moodle, where the Moodle server does not need to store the duration directly. The user-duration process and storage load is completely left to the Log aggregator.

Author Contributions

All authors have equal contributions to the paper. All the authors have read and approved the final manuscript.

Funding

No funding support was received.

Declaration of Conflicting Interests

The author declared no potential conflicts of interest with respect to the research, authorship, and/ or publication of this article.

Data Availability

The datasets generated during and/ or analysed during the current study are available from the corresponding author on reasonable request.

Declaration on AI Use

The authors declare that no artificial intelligence (AI) or AI-assisted tools were used in the preparation of this manuscript. AI were used only to improve readability and language under strict human oversight; no content, ideas, analyses, or conclusions were generated by AI.

References

[1] A. Badia, D. Martín, and M. Gómez, "Teachers' Perceptions of the Use of *Moodle* Activities and Their Learning Impact in Secondary Education," Technology, Knowledge and Learning, vol. 24, no. 3, pp. 483–499, 2019, doi: 10.1007/s10758-018-9354-3.

[2] G. v. Davidson-Shivers, K. L. Rasmussen, and P. R. Lowenthal, Web-based learning: Design, implementation and evaluation: Second edition. 2017. doi: 10.1007/978-3-319-67840-5.

[3] E. E. Egorov, M. P. Prokhorova, T. E. Lebedeva, O. A. Mineeva, and S. Y. Tsvetkova, "*Moodle* LMS: Positive and Negative Aspects of Using Distance Education in Higher Education Institutions," Propósitos y Representaciones, vol. 9, no. SPE2, 2021, doi: 10.20511/pyr2021.v9nspe2.1104.

[4] D. Monllaó, "Analytics," 2019. https://docs.*Moodle* .org/35/en/Analytics (accessed May 21, 2022).

[5] H. Foster, "Logs," 2020. https://docs.*Moodle* .org/400/en/Logs (accessed May 21, 2022).

[6] M. Manzo, A. Petrosino, A. Maratea, and A. Ferone, "Tracking *e-learning* user *session* s," no. December, 2013, doi: 10.13140/RG.2.1.3789.8723.

[7] M. Cooch, "*Page* Resources," 2022. https://docs.*Moodle* .org/400/en/*Page* _resource (accessed May 21, 2022).

[8] E. Zacarias et al., "Optimizing the Access Records of Students in the *Moodle* Virtual Learning Environment Database," IFAC-PapersOnLine, vol. 49, no. 30, pp. 98–101, 2016, doi: 10.1016/j.ifacol.2016.11.135.

[9] MDN contributors, "*AJAX* Getting Started," 2022. https://developer.mozilla.org/en-US/docs/Web/Guide/*AJAX*/Getting_Started (accessed May 21, 2022).

[10] T. Turc, "*AJAX* Technology for Internet of Things," in Procedia Manufacturing, 2019, vol. 32, pp. 613–618. doi: 10.1016/j.promfg.2019.02.260.

[11] A. Nicols, E. Lafuente, D. Mudrak, and H. Foster, "Coding style," https://docs.*Moodle* .org/dev/Coding_style, Apr. 11, 2022. https://docs.*Moodle* .org/dev/Coding_style (accessed Nov. 19, 2022).

[12] R. T. Fielding, "Architectural Styles and the Design of Network-based Software Architectures," 2000.

[13] L. Richardson, M. Amundsen, and S. Ruby, *REST* ful Web *API* s, 1st ed. Sebastopol: O'Reilly Media, Inc, 2013.

[14] A. Borrego, M. Bermudo, F. Sola, D. Ayala, I. Hernández, and D. Ruiz, "Silence — A web framework for an agile generation of *REST ful API* s," SoftwareX, vol. 20, p. 101260, Dec. 2022, doi: 10.1016/j.softx.2022.101260.

[15] D. Mudrak et al., "Local plugins," May 06, 2022. https://docs.*Moodle* .org/dev/Local_plugins (accessed Nov. 25, 2022).

[16] "*Time* is money. Know how you spend it online." https://*time*yourweb.com/ (accessed Jan. 17, 2023).