

Integration between Moodle and Academic Information System using Restful API for Online Learning

Novian Adi Prasetyo¹, Yudha Saintika²

¹Program Studi S1 Teknik Informatika, Institut Teknologi Telkom Purwokerto

²Program Studi S1 Sistem Informasi, Institut Teknologi Telkom Purwokerto

ARTICLE INFO

Article history:

Received September 24, 2021

Revised October 11, 2021

Accepted October 20, 2021

Keywords:

Academic Information System;

Learning Management System;

Moodle;

Restful API;

Synchronization

ABSTRACT

During the current pandemic, it is encouraging educational institutions to carry out distance learning, so many learning management system (LMS) platforms can be used to support distance learning. Each LMS has a different process flow but has the same goal of making it easier to manage learning content. When an LMS is implemented in an educational institution, it requires matching data for courses, students and lecturers that are available in the academic information system (AIS) at the institution, this is one of the weaknesses of all LMS because the data are not interrelated between AIS and LMS. The purpose of this research is to create an integrated system to equalize data between AIS and LMS using the synchronization method through the Application Programming Interface (API). The results of this application will combine data from AIS and LMS which will then be tested for automatic course creation according to class data, courses, lecturers and students at AIS. The test results of this system are said to be successful because each function that is designed has been running well without any fatal errors. The most important thing that needs to be considered when synchronizing is that there is a link between the data on the AIS and LMS, failure occurs on some courses because the email users in the AIS and LMS are different.

This work is licensed under a [Creative Commons Attribution-Share Alike 4.0](https://creativecommons.org/licenses/by-sa/4.0/)



Novian Adi Prasetyo,

Institut Teknologi Telkom Purwokerto, Jl. DI Panjaitan No.128, Karangreja, Purwokerto Selatan 53147, Indonesia

Email: novian@ittelkom-pwt.ac.id

1. INTRODUCTION

During a pandemic that occurs around the world, this causes all agencies to carry out work from home activities, as well as educational institutions in Indonesia that implement learning from home, this method is called distance learning [1][2]. Distance learning in all educational institutions is are required to prepare all infrastructure to simplify the distance learning process carried out by lecturers and students, with the support of newest technology in distance learning can be done easily using several applications that are indeed developed for the process of implementing distance learning and learning management system (LMS) [3], some of the popular LMSs are Moodle [4], Chamilo, Open edX Studio, Totara Learn, Canvas [5].

Learning management system is a learning technology that can be used to carry out distance learning in universities [6]. Each LMS has a different architecture depending on the respective developers such as Moodle [7], Sakai, ATutor is an LMS that is Open Source, such as Blackboard, Success Factors, Sum Total is a commercial LMS [8]. In a study evaluating the features of the Moodle LMS. Moodle is designed to be able to present learning content, development of lesson plans, assessments, learning feedback, communication between students and teachers. Some of the most used features on Moodle are assignment collection, feedback, and quizzes [9].

Academic information system (AIS) is a technology for data storage and management in educational institutions [10]. Each educational institution has an AIS with its own needs, the development of AIS is carried out individually so that the business processes expected by the institution can match the exact needs [11]. Apart from storing all academic data, AIS can also be used to predict the academic abilities of students in the future

[12]. AIS has an extensive data collection, student data management is a major factor that must be considered because an educational institution should have hundreds or even thousands of students, with the AIS running according to the procedure, all learning needs can be controlled under proper monitoring [11]. Based on research the AIS has implemented a lot of web bases, so that AIS is more flexible in its use and can be easily synchronized with other applications.

Application Programming Interface (API) is a web service that can be used to connect the client and server [13], API can be developed in two ways, namely Simple Object Access Protocol (SOAP) and Representational State Transfer (REST) [14]. The use of REST has the advantage of lower bandwidth consumption compared to SOAP, in this study [15] testing REST and SOAP to see the amount of bandwidth consumption in several processes (auth, login and getGrades), the result is that the bandwidth consumption on REST is 73% smaller compared to SOAP.

From the above explanation, several existing technologies can be used for learning management in educational institutions, with the implementation of distance learning [16], learning technology must be improved, in this study the Moodle-based LMS was used as a learning media and AIS belonging to one of the higher education institutions, the AIS was determined. The data that will be used in this research are class data, lecturers, students and subjects. With all the advantages that Moodle has, it is found that a bootle neck will be experienced by all types of LMS, namely related to user data in it. With the AIS that is already available in educational institutions, this will help in obtaining overall student data. By using a primitive method, students will be led to do independent enroll according to the courses taken [17]. The lecturers will also make the courses independently according to their respective learning styles [17]. So far this has not been a problem because distance learning is still running.

As time goes by, the data will continue to grow more and more, with no rules in making courses in the LMS this will cause data redundancy. Data redundancy can occur because each year the same course will be made but the lecturers and students in it must be different, so it is impossible to delete old users or create the same lesson name. Things that can be used to avoid this problem are the instructions given to the teachers so that the courses are made in accordance with the instructions [18]. Another problem found in using Moodle is that many users have to do enroll independently, the independent enroll done by students has problems in the form of difficulty finding the name of the lesson that matches the course they are currently taking.

Based on observations on Moodle used in this study, many lecturers did not understand if there were students who had not entered the course, many subjects had many classes while the material inside was the same, if all classes were combined in one course in LMS, the lecturer would have difficulty identifying students in each class. The concept of team teaching will also experience problems if the courses in the LMS are combined because each lecturer cannot identify the students who are members based on their class. Moodle has a group that can be used to group students, so that with this group the lecturer can easily identify each student [19]. However, with the number of hundreds of students in each course it will take a lot of time to do the grouping.

Based on the above problems, an LMS data integration solution is taken [20][21] with higher education AIS to facilitate class data management, lecturers and students at LMS use data sources from AIS. The RESTful Web API will be used in this synchronization process, where Moodle already has a RESTful Web API that can be used in open source, while the AIS of the related university is currently web-based so that communication between these two systems can be done easily.

2. METHOD

In this research, the method used is the synchronization method [22] by combining heterogeneous data [23], namely using a way of linking data between AIS and LMS through the application programming interface (API) on the two systems. An integration semi structured and unstructured data [24] system will be built that can accommodate data from AIS and LMS, the integration system will have the ability to control the process of creating read update delete courses without having to open the LMS.

2.1. RESTful Web Application Programming Interface

RESTful Web API is the main infrastructure used for the data integration process [25] between AIS and LMS, the architecture for using the RESTful Web API in AIS and LMS can be seen in Fig. 1. As shown in Fig. 1, the API used is sourced from AIS and LMS, the integration system will make HTTPS requests [26] to the AIS and LMS then get a response in the form of JavaScript Object Notation (JSON) and integrate it into the integration system database [27]. Table 1 is a list of the Uniform Resource Identifier (URI) of the AIS. Table 2 is a list of the URI of the LMS.

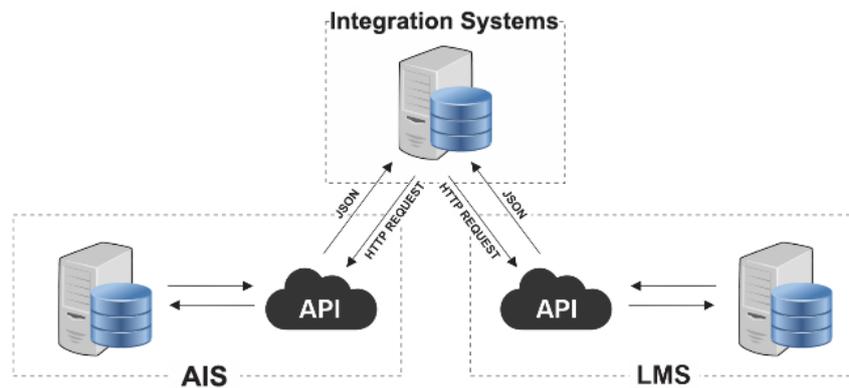


Fig. 1. RESTful Web API Architecture on System Integration

Table 1. URI List on AIS

Name	Response Description
GET Study Program	Issue data responses in the form of JSON in the form of a list of study programs at AIS.
GET Course in Study Program	Issue data responses in the form of JSON in the form of a list of courses based on the selected study program.
GET Student in Course	Issue data responses in the form of JSON in the form of a student list based on the selected course.

Table 2. List of URI in LMS

Name	Response Description
GET Courses by field	Output the data response in the form of an array in the form of detailed courses in the LMS.
GET Categories	Output the data response in the form of an array in the form of a list of course categories. The categories in the LMS are used as study programs.
POST Create Courses	Receive data in the form of an array to instruct the loading of new courses in the LMS.
POST Import Course	Receiving data in the form of an array to instruct the process of importing the template course to the course that was just created.
GET Course Groups	Output the response data in the form of an array in the form of a list of course groups, this group is used to group the classes in each course.
POST Create Groups	Receive data in the form of an array to instruct the group creation in the course.
POST Manual Enroll Users	Receive data in the form of an array to instruct user enroll to the course using email.
POST Add Group Members	Receive data in the form of an array to instruct users to enter the group / course class.
GET Enrolled Users	Outputs data in the form of an array in the form of a list of registered users in the course.
POST Manual Unenrolled Users	Receive data in the form of an array to command the user unenrolled in the course.
POST Delete Courses	Receiving data in the form of an array to order deleting courses in the LMS.
GET Contents	Outputs data in the form of an array in the form of detailed content in the selected course.
GET Course Module	Outputs data in the form of an array in the form of a list of modules on the selected content.

2.2. Short Name Generate

In this section the integration system will automatically generate a short name using the standard format. Standardization in making short names is made in the format shown in Fig. 2. Short name format as in Fig. 2

is unique so that the generated results will not be the same as other courses, this is very important for the integration system so that it can link AIS data and LMS data, so that each course can be synchronized with each other according to the existing content, lecturers and students.

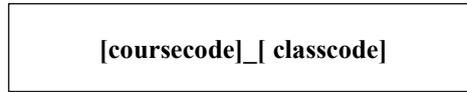


Fig. 2. Short name Standard Format

2.3. Data Synchronization Flowchart

The synchronization process is carried out by saving data from AIS and from the LMS to the integration system database, in Fig. 3 is a data synchronization process flow that must be carried out sequentially so that data can be generated properly without data redundancy [28].

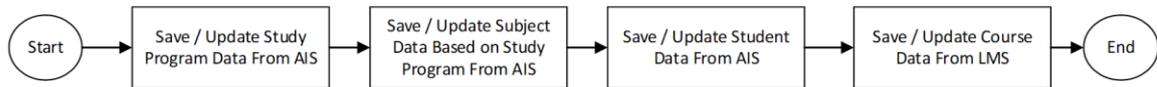


Fig. 3. Data Synchronization Flowchart

2.4. Create Course Flowchart

The process of creating an LMS course is carried out after the data synchronization process has been carried out properly, this is required so that the data on the resulting course is in accordance with AIS's data. Fig. 4 presents a flowchart of the course creation process from the integration system to the LMS.

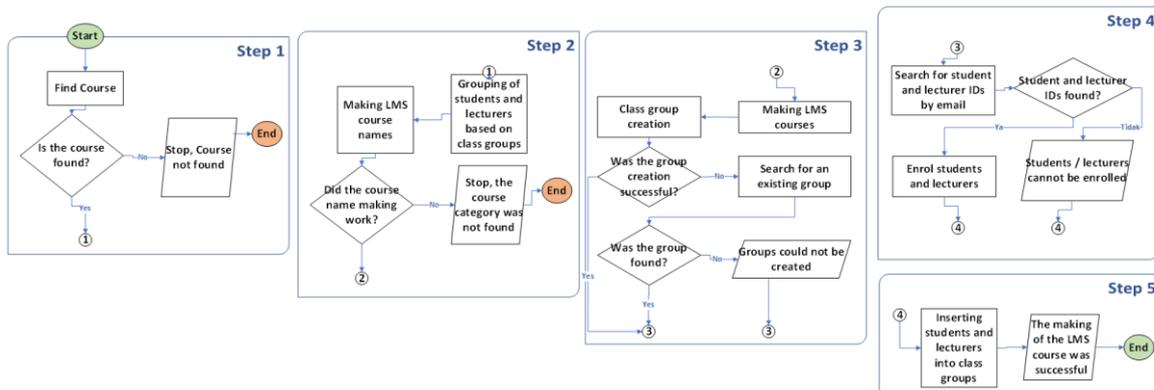


Fig. 4. Create Data Synchronization Flowchart

Fig. 4 shows that there are five main steps in making an LMS course. Fig. 4 (Step 1) is the course validation step to ensure that the course parameters to be made are available at AIS, if the course is not found, the process will stop and fail. Fig. 4 (Step 2) is a step to group student and lecturer accounts and create a shortname as described in Section 2.2, in making course names, the course category is required to be available in the LMS, if these conditions are not met, the process will stop and fail. Fig. 4 (Step 3) is the step of making courses and making class groups in LMS with branching conditions which can be seen in Fig. 4 (Step 3), if the conditions are not met, the class group will not be created. Fig. 4 (Step 4) is the stage of searching for an LMS user ID based on email, the user ID that has been found will be sent to the course currently being created, if the ID is not found then the user will not enter the course. Fig. 4 (Step 5) is the stage of entering each user who has successfully enrolled in Fig. 4 (Step 4) into the class groups that were created in Fig. 4 (Step 3).

2.5. User LMS Update Flowchart

After the LMS create course is successfully carried out, there is an optional process, namely the update course, this process aims to re-synchronize the LMS user if there are changes in lecturers, students and classes at AIS. This process flow diagram can be seen in Fig. 5.

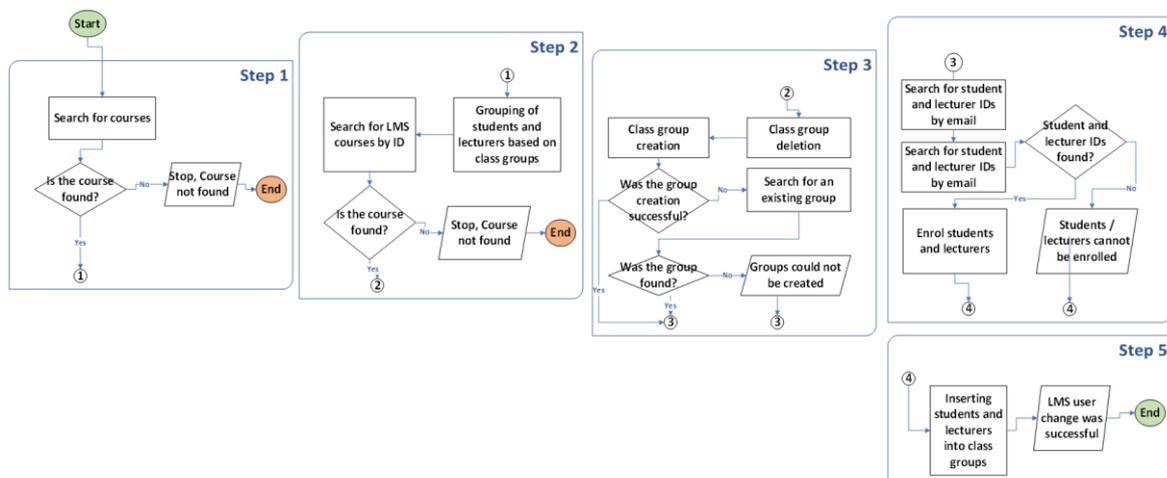


Fig. 5. Update Data Synchronization Flowchart

Fig. 5 shows that there are five main steps in changing the LMS user course. Fig. 5 (Step 1) is the course validation step to ensure that the course parameters to be changed are available in AIS, if the course is not found, the process will stop and fail. Fig. 5 (Step 2) is the step of grouping student and lecturer accounts and searching for LMS courses based on ID, if these conditions are not met then the process will stop and fail. Fig. 5 (Step 3) is the step of deleting class groups and creating new class groups in LMS with branching conditions which can be seen in Fig. 5 (Step 3), if the conditions are not met, the class group will not be created. Fig. 5 (Step 4) is the stage of un-enrolling the account in the course then the process of searching for the LMS user ID based on the email is carried out, the user ID that has been found will be emailed back to the course that is currently being changed, if the ID is not found then the user will not be entered into courses. Fig. 5 (Step 5) is the stage of entering each user who has been successfully enrolled in Fig. 5 (Step 4) into the class groups that were created in Fig. 5 (Step 3).

3. RESULTS AND DISCUSSION

3.1. System Implementation

The integrated system has been successfully created with various features that have been designed. The first thing to pay attention to is that student, lecturer and class data must be well synchronized. Fig. 6 shows the data of students, lecturers and classes that have been successfully synchronized with the integrated system. Fig. 6 (A) describes a page of student data that has been synchronized from AIS into the integration system, Fig. 6 (B) describes the lecturer data page that has been synchronized from AIS into the integration system, Fig. 6 (C) describes the class data page that has been synchronized from AIS into the integration system. Based on the validation with the AIS data in Fig. 6 is stated that it is well synchronized, that is, without data redundancy.

The next thing that needs to be considered is the results of class, lecturer and student data processing into course data. Fig. 7 displays the subject data resulting from combining student, lecturer and class data. Fig. 7 describes the course data page, on that page there are 3 buttons that can be executed to create, change, read and delete courses in the LMS. The create course button is used to create a course as described in Section 2.4, the result of the create course can be seen in Fig. 8. The update course button is used to change the course as described in Section 2.5. The course detail button is used to read the content of the course in the LMS so that the course content can be seen directly without having to open the LMS. The delete button is used to delete courses in the LMS.

Fig. 8 describes the results of the create course that has been carried out from the integration system, it appears that there is a short name that has been successfully created automatically using the short name generator as described in Section 2.2.

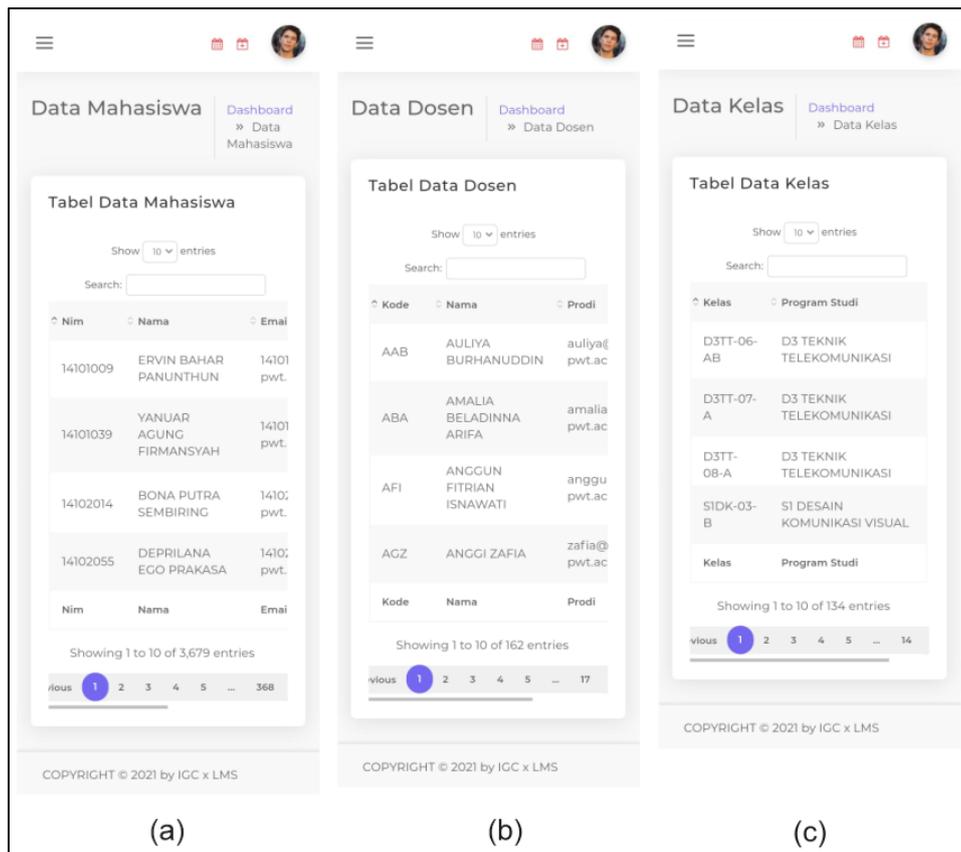


Fig. 6. Data for Students (A), Lecturer (B) and Class (C) for the Odd 2020/2021 Academic Year

DTS305222	PRAKTEK KERJA LAPANGAN	D3 TEKNIK TELEKOMUNIKASI	D3TT-06-AB (54) - NURLAILI	Create Course
DTS305324	TUGAS AKHIR	D3 TEKNIK TELEKOMUNIKASI	D3TT-06-A (31) - MUNTAQO ALFIN AMANAF	Create Course
DTS314102	NETWORK SECURITY	D3 TEKNIK TELEKOMUNIKASI	D3TT-06-A (20) - SYARIFUL IKHWAN D3TT-06-B (5) - SYARIFUL IKHWAN	Detail Course Delete Course Update Course
DTS314202	KOMUNIKASI RADIO GELOMBANG MIKRO	D3 TEKNIK TELEKOMUNIKASI	D3TT-06-A (27) - PRASETYO YULIANTORO D3TT-06-B (40) - PRASETYO YULIANTORO	Detail Course Delete Course Update Course

Fig. 7. Course Data

DK6352903_S1DK02 DESAIN KEMASAN (S1DK 02A 02B 02C)

Teacher: Emmareta Fauziah, Faradilla Intan Kusumadewi, Krishnanda Mooduto

Click to enter this course

Category: S1 DESAIN KOMUNIKASI VISUAL

Fig. 8. Course Views After the Create Course is Running

Fig. 9 describes the course detail page which can be seen from the integration system, the data that appears on that page is taken based on the data in the LMS, so that the monitoring process can be carried out directly through the integration system. Fig. 10 is a detailed view of the LMS, if a comparison is made between Fig. 9 with Fig. 10 can be seen that each contains the same data. Fig. 11 describes class data, lecturers and students. If a comparison is made between Fig. 11 with Fig. 9 can be seen that the class has the same number, namely 3, the lecturers have the same number, namely 3, the students have the same number, namely 94. The data will certainly amount to the same because the data in Fig. 11 and Fig. 9 were both taken from the same source,

namely from the LMS. Fig. 12 describes class data, lecturers semester and students. If a comparison is made between Fig. 12 with Fig. 11 can be seen that the class has the same number, namely 3, the lecturers have the same number, namely 3, while the students have the difference in the number of 3 students. From the results of the analysis, the difference between 3 students was due to new additions made by the lecturer because there was a sudden change in the study plan card.

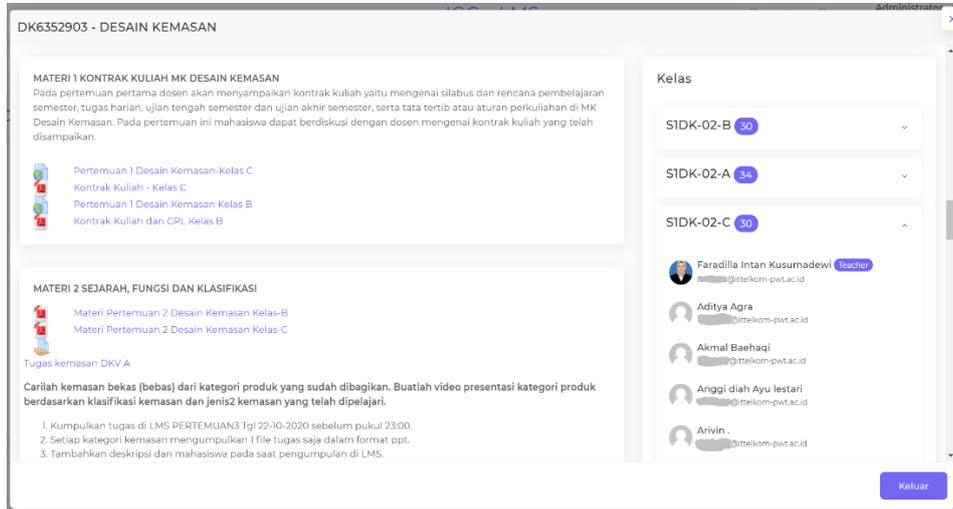


Fig. 9. System Integration Course Details

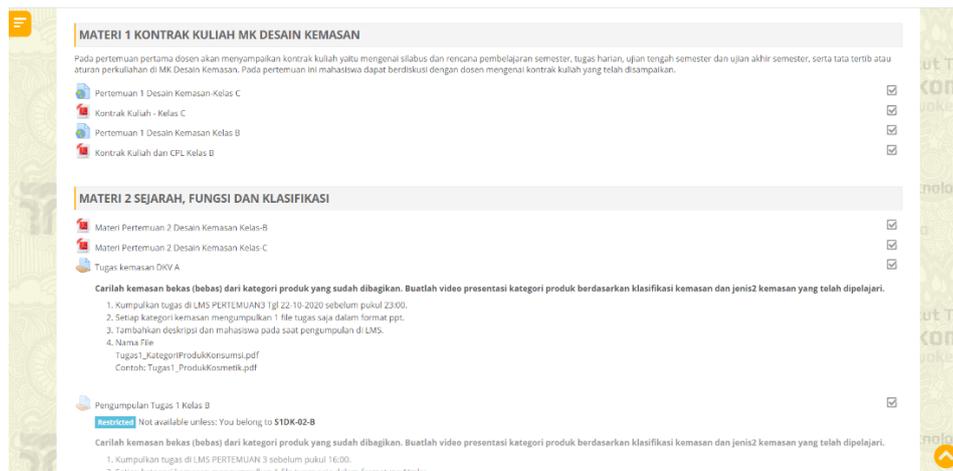


Fig. 10. LMS Course Details

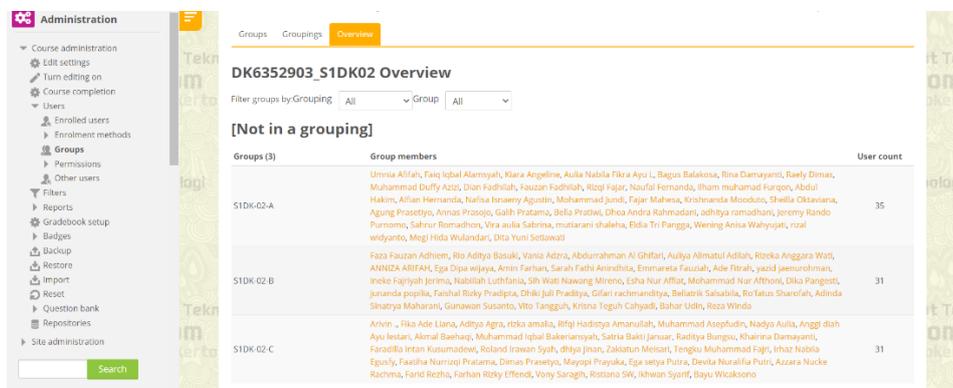


Fig. 11. Class, Lecturer and Student (LMS) Data

The screenshot shows a web application interface for 'Data Matakuliah'. It features a search bar with the text 'DESAIN KEMASAN'. Below the search bar is a table with the following columns: Code, Nama, Prodi, Kelas, and Aksi. The table contains one row of data: Code: DK6352903, Nama: DESAIN KEMASAN, Prodi: SI DESAIN KOMUNIKASI VISUAL, Kelas: SIDK-02-A (34) - KRISHNANDA, RADITYA MOODUTO, SIDK-02-B (27) - EMMARETA FAUZIAH, SIDK-02-C (30) - FARADILA INTAN, KUSUMADEWI. The Aksi column contains three buttons: 'Detail Course' (blue), 'Delete Course' (red), and 'Update Course' (purple). At the bottom of the table, it says 'Showing 1 to 1 of 1 entries (filtered from 335 total entries)'. There are also navigation buttons for 'Previous' and 'Next'.

Fig. 12. Class, Lecturer and Student Data (AIS)

3.2. System Testing

At this stage the system testing will be carried out to ensure that the system being developed is running according to its functionality [29], testing this system using blackbox testing as a means of validating existing functionality [30], blackbox testing is internal testing at the development stage, so testing is carried out by internal developers until the feature runs according to the expected results. Blackbox testing results can be seen in Table 3.

Table 3. Functional Testing Results

Features	Expectation	Observation	Result
Synchronization of Study Program Data	Can update study program data on the integration system according to AIS	The study program data was successfully updated, the results are in accordance with the data in AIS	Succeed
Data Synchronization of AIS Subject	Can update study program data on the integration system according to AIS	The course data was successfully updated, the results match the data in AIS	Succeed
Synchronization of Data for LMS Subjects	Can update study program data on the integration system according to AIS	The course data was successfully updated, the results match the data in the LMS	Succeed
Synchronization of Students Based on Study Programs Dashboard	Can update students data on the integration system according to AIS	Student data was successfully updated, the results match the data in AIS	Succeed
	1. Can display the comparison of courses on AIS and LMS correctly	1. Comparative data on AIS and LMS subjects are displayed according to the stored data	Succeed
	2. Can display recap data from Class, Lecturers and Students that have been integrated into the integration system correctly	2. Class recap data, lecturers and students are displayed according to the data stored	
Data Dosen	Can display a list of lecturers that have been integrated	Lecturer data appear accordingly	Succeed
Data Matakuliah	Can display a list of courses that have been integrated	Subject data appear accordingly	Succeed
Data Kelas	Can display a list of classes that have been integrated	Class data appears accordingly	Succeed
Data Mahasiswa	Can display a list of students who have been integrated	Student data appears accordingly	Succeed
Create Course	1. Can make courses in the LMS through the integration system 2. Can generate a short name 3. Can create groups for classes 4. Can send lecturers and students to class groups	1. Course on the LMS was successfully created 2. Short name generated according to function 3. Class groups are made according to the name and number 4. Lecturers and students can enroll according to the number, there are some students who are not registered because the causes of email on AIS and LMS are different	Succeed
Detail Course	Can read content and LMS users through the integration system	Content and the number of users on the LMS can be seen through the integration system	Succeed
Update Course	Can update class, lecturer and student data according to the integrated system without losing LMS content data	Class groups, lecturers and students were successfully updated without losing content on the LMS	Succeed
Delete Course	Can delete courses in the LMS through the integration system	Courses on the LMS are deleted via the integration system	Succeed

4. CONCLUSION

The integration system has been successfully created and functionality testing is fully successful. The data synchronization process from AIS to the integration system runs well without any fatal errors, the data synchronization process from the integration system to the LMS runs well without any fatal errors. There were several mistakes that occurred during the course creation process, namely that not all students were enrolled as according to the data in AIS, this happened because there was a human error which caused the email on AIS and LMS to be different. So it can be concluded that the integration between AIS and LMS has been successfully carried out. In its implementation, this integration system has been able to make courses through the integration system 319 times without any fatal errors. The course making process takes the longest is 5 minutes, this is because there is a course that has a large number of students, this directly affects the speed. The resulting course is in accordance with the specified template, which will uniform the content of all existing courses. This integration still needs to be further developed, so that the learning system using LMS can be done from one door. Some of the main things that still need to be developed are attendance integration, schedule integration and content integration.

Acknowledgments

This research was supported by the Institut Teknologi Telkom Purwokerto as a funder and the Institut Teknologi Telkom Purwokerto's Information System Unit as a data provider.

REFERENCES

- [1] S. Syahrudin, M. F. Mohd Yaakob, A. Rasyad, A. W. Widodo, S. Sukendro, S. Suwardi, A. Lani, L. P. Sari, M. Mansur, R. Razali, and A. Syam, "Students acceptance to distance learning during Covid-19: the role of geographical areas among Indonesian sports science students," *Heliyon*, vol. 7, no. 9, p. e08043, 2021. <https://doi.org/10.1016/j.heliyon.2021.e08043>
- [2] E. M. Zamzami, "Aplikasi Edutainment Pendukung Pembelajaran Jarak Jauh TK Merujuk Standar Nasional PAUD," *J. Obs. J. Pendidik. Anak Usia Dini*, vol. 5, no. 2, pp. 985–995, 2021. <https://doi.org/10.31004/obsesi.v5i2.750>
- [3] A. Hutchison, "Technological Efficiency in The Learning Management System: A Wicked Problem with Sustainability for Online Writing Instruction," *Comput. Compos.*, vol. 54, p. 102510, 2019. <https://doi.org/10.1016/j.compcom.2019.102510>
- [4] M. D. Dascalu *et al.*, "Before and during COVID-19: A Cohesion Network Analysis of students' online participation in moodle courses," *Comput. Human Behav.*, vol. 121, no. October 2020, p. 106780, 2021. <https://doi.org/10.1016/j.chb.2021.106780>
- [5] C. Pappas, "The Best Learning Management Systems (2022 Update)," *eLearning Industry*, 2022. [Online]. Available: <https://elearningindustry.com/the-best-learning-management-systems-top-list>. [Accessed: 20-Oct-2022]
- [6] I. Almarashdeh, "Sharing instructors experience of learning management system: A technology perspective of user satisfaction in distance learning course," *Comput. Human Behav.*, vol. 63, pp. 249–255, 2016. <https://doi.org/10.1016/j.chb.2016.05.013>
- [7] S. F. Shetu, M. M. Rahman, A. Ahmed, M. F. Mahin, M. A. U. Akib, and M. Saifuzzaman, "Impactful e-learning framework: A new hybrid form of education," *Curr. Res. Behav. Sci.*, vol. 2, no. March, p. 100038, 2021. <https://doi.org/10.1016/j.crbeha.2021.100038>
- [8] N. N. M. Kasim and F. Khalid, "Choosing the right learning management system (LMS) for the higher education institution context: A systematic review," *Int. J. Emerg. Technol. Learn.*, vol. 11, no. 6, pp. 55–61, 2016. <https://doi.org/10.3991/ijet.v11i06.5644>
- [9] K. C. Deepak, "Evaluation of Moodle Features at Kajaani University of Applied Sciences-Case Study," *Procedia Comput. Sci.*, vol. 116, pp. 121–128, 2017. <https://doi.org/10.1016/j.procs.2017.10.021>
- [10] S. Mehroliya, S. Alagarsamy, and M. Indhu Sabari, "Moderating effects of academic involvement in web-based learning management system success: A multigroup analysis," *Heliyon*, vol. 7, no. 5, p. e07000, 2021. <https://doi.org/10.1016/j.heliyon.2021.e07000>
- [11] L. I. Sarhan, A. M. Atroshi, and N. S. Ahmed, "A Strategic Planning of Developing Student Information Management System Using SWOT Technique," *J. Univ. Hum. Dev.*, vol. 2, no. 3, p. 515, 2016. <https://doi.org/10.21928/juhd.v2n3y2016.pp515-519>
- [12] A. A. Saa, M. Al-Emran, and K. Shaalan, *Mining Student Information System Records to Predict Students' Academic Performance*, vol. 921, Springer International Publishing, 2020. https://doi.org/10.1007/978-3-030-14118-9_23
- [13] A. Zakaria, A. Prihantara, and A. A. Hartono, "Integrasi Application Programming Interface, PHP, dan MySQL untuk Otomatisasi Verifikasi dan Aktifasi Pengguna Layanan Hotspot MikroTik," *JUITA J. Inform.*, vol. 7, no. 2, p. 63, 2019. <https://doi.org/10.30595/juita.v7i2.4361>
- [14] S. Mowla and S. V. Kolekar, "Development and integration of E-learning services using rest APIs," *Int. J. Emerg. Technol. Learn.*, vol. 15, no. 4, pp. 53–72, 2020. <https://doi.org/10.3991/ijet.v15i04.11687>
- [15] C. M. Garcia and R. Abilio, "Systems Integration Using Web Services, REST and SOAP: A Practical Report," *Sistemas*, vol. 19, pp. 34–41, 2017. http://www.fsma.edu.br/si/edicao19/FSMA_SI_2017_1_Principal_3_en.pdf
- [16] T. D. Oyedotun, "Sudden change of pedagogy in education driven by COVID-19: Perspectives and evaluation from

- a developing country,” *Res. Glob.*, vol. 2, no. November, p. 100029, 2020. <https://doi.org/10.1016/j.resglo.2020.100029>
- [17] S. V. Kolekar, R. M. Pai, and M. M. Manohara Pai, “Adaptive User Interface for Moodle based E-learning System using Learning Styles,” *Procedia Comput. Sci.*, vol. 135, pp. 606–615, 2018. <https://doi.org/10.1016/j.procs.2018.08.226>
- [18] J. A. Kumar and H. Al-Samarraie, “MOOCs in the Malaysian higher education institutions: The instructors’ perspectives,” *Ref. Libr.*, vol. 59, no. 3, pp. 163–177, 2018. <https://doi.org/10.1080/02763877.2018.1458688>
- [19] R. F. Bataineh and M. B. Mayyas, “The utility of blended learning in EFL reading and grammar: A case for moodle,” *Teach. English with Technol.*, vol. 17, no. 3, pp. 35–49, 2017. <https://www.ceeol.com/search/article-detail?id=551005>
- [20] K. Dobashi, “Automatic data integration from Moodle course logs to pivot tables for time series cross section analysis,” *Procedia Comput. Sci.*, vol. 112, pp. 1835–1844, 2017. <https://doi.org/10.1016/j.procs.2017.08.222>
- [21] S. Sucipto and J. Karaman, “Integration of Legalization Information System Web-Based using Shipping API and Telegram API,” *JUITA J. Inform.*, vol. 8, no. 2, p. 131, 2020. <https://doi.org/10.30595/juita.v8i2.7104>
- [22] F. Boronat, M. Montagud, P. Salvador, and J. Pastor, “Wersync: A web platform for synchronized social viewing enabling interaction and collaboration,” *J. Netw. Comput. Appl.*, vol. 175, p. 102939, 2021. <https://doi.org/10.1016/j.jnca.2020.102939>
- [23] F. Xu, Z. Wu, F. Wen, and J. Tang, “Design and Implementation of a Heterogeneous Database Synchronization Mechanism Based on the Netgap,” *Proc. - 2018 5th Int. Conf. Inf. Sci. Control Eng. ICISCE 2018*, pp. 181–186, 2019. <https://doi.org/10.1109/ICISCE.2018.00047>
- [24] A. C. Anadiotis, O. Balalau, C. Conceição, H. Galhardas, M. Y. Haddad, I. Manolescu, T. Merabti, and J. You, “Graph integration of structured, semistructured and unstructured data for data journalism,” *Information Systems*, p. 101846, 2021. <https://doi.org/10.1016/j.is.2021.101846>
- [25] R. Padmanaban, M. Thirumaran, P. Anitha, and A. Moshika, “Computability evaluation of RESTful API using Primitive Recursive Function,” *Journal of King Saud University - Computer and Information Sciences*, 2018. <https://doi.org/10.1016/j.jksuci.2018.11.014>
- [26] E. S. Alashwali, P. Szalachowski, and A. Martin, “Exploring HTTPS security inconsistencies: A cross-regional perspective,” *Comput. Secur.*, vol. 97, 2020. <https://doi.org/10.1016/j.cose.2020.101975>
- [27] P. Bourhis, J. L. Reutter, and D. Vrgoč, “JSON: Data model and query languages,” *Information Systems*, vol. 89, p. 101846, 2021. <https://doi.org/10.1016/j.is.2019.101848>
- [28] T. Andriyanto and R. Indriati, “RESTful Web Service Untuk Integrasi Data Antar Sistem Informasi,” *J. Sains dan Inform.*, vol. 3, p. 116, Nov. 2017. <https://doi.org/10.34128/jsi.v3i2.111>
- [29] S. Supriyono, “Software Testing with the approach of Blackbox Testing on the Academic Information System,” *Int. J. Inf. Syst. Technol.*, vol. 3, no. 2, pp. 227–233, 2020. <http://ijistech.org/ijistech/index.php/ijistech/article/view/54>
- [30] V. Garousi, A. Rainer, P. Lauvås, and A. Arcuri, “Software-testing education: A systematic literature mapping,” *J. Syst. Softw.*, vol. 165, 2020. <https://doi.org/10.1016/j.jss.2020.110570>

BIOGRAPHY OF AUTHORS



Novian Adi Prasetyo is a lecturer with expertise in web application development and mobile application development. The main research topics are web and mobile applications that are integrated with other systems, applications that have intelligent systems and enterprise applications. Email: novian@ittelkom-pwt.ac.id



Yudha Saintika is a lecturer with expertise in Information System, e-Government, Smart City, e-Learning and IT Adoption. Email: yudha@ittelkom-pwt.ac.id