



Development of the RPS Module in the Portal-Prodi System at the University of Lampung using the RAD Method

Mahendra Pratama[✉], Rivan Herdian Pramudya, Mona Arif Muda, Hery Dian Septama

Department of Electrical Engineering, Faculty of Engineering, Universitas Lampung, Bandar Lampung, Indonesia

Article Info

Article History:

Received: 26 July 2024

Revised: 28 August 2024

Disetujui: 30 August 2024

Keywords:

Higher Education Administration, Rapid Application Development, Semester Learning Plans

Abstract

The development of the RPS Module in the University of Lampung Study Program Portal System aims to streamline and standardize the creation of Semester Learning Plans (RPS) in various study programs. In 2020, the ministry issued a policy ranking state universities based on key performance indicators (KPI). The KPI assessment has eight indicators, including KPI 7: Collaborative and Participatory Class. In this indicator, the Semester Learning Plan (RPS) is one of the main pieces of evidence for universities stating that lectures taught by lecturers are carried out using the Project Based Method or Case-Based Method. This research uses the Rapid Application Development (RAD) methodology. The iterative nature of the RAD method facilitates efficient requirements gathering, user design, construction, and implementation stages. The main features of the semester learning plan module have been successfully developed based on the features that have been created, namely, submitting RPS, validating RPS, viewing RPS, and printing RPS by generating RPS into a PDF file. Based on testing the semester learning plan module using the black box method, the test results followed the expected results and went well. This study contributes to ongoing efforts to integrate digital solutions in higher education administration.

INTRODUCTION

In 2020, COVID-19 spread throughout the world, and this caused services that were previously offline to become entirely online due to the need to comply with health protocols (Girum et al., 2020). So, an application called Portal-Prodi was developed to facilitate administration at the study program level, especially in the electrical engineering department at the University of Lampung (UNILA) (Pradipta et al., 2021). This system develops according to needs using a top-down approach, starting from the needs of the national level, university leaders, faculty leaders, lecturers, and students. This research aims to develop the Portal Prodi to add a Learning Semester Plan Module.

In the same year, the ministry issued a policy regarding ranking state universities based on key performance indicators (KPI) (Mendikbud, 2020). The KPI assessment has eight indicators, including KPI 7: Collaborative and Participatory Classes. In this indicator, the Semester Learning Plan (RPS) is one of the primary pieces of evidence for universities claiming that lecture classes taught by lecturers are run using the Project-Based Method or Case-Based Method.

The biggest problem with collecting RPS is that most lecturers don't know the format and how to fill in the RPS, which makes the RPS templates for each study program different. Therefore, in this research, a module was created to help course lecturers prepare RPS according to the format determined at UNILA.

A library is used in this journal to explain the technology that will be used in this journal. Include the following:

A. Web Service

Web Service is a system designed to support activities between systems on a network. Web service is an interface that contains a collection of operations that can be accessed by a network, for example, via the internet, with XML (eXtensible Markup Language) format. Web Service consists of a collection of functions and methods centered on a server that users can call, where developers can access 10 of these methods even when using different programming languages and platforms (Marthasari et al., 2012).

B. PostgreSQL

Postgresql is a database developed by the University of California at Berkeley Computer Science Department. Many platforms support Postgresql, and it is free of charge. Postgresql is also a relational database that supports SQL commands. Postgresql also has complete documentation, a large community, and is

supported by various programming languages such as Java, PHP, Python, C ++, etc. (Prasetyo, 2022).

C. Laravel

Laravel is an open-source PHP framework with the Model-View-Controller (MVC) design used to build website applications. Taylor Otwell first built this framework on February 22, 2012. PHP is usually used on the backend to develop dynamic websites. With Laravel, website development can run faster and simpler (Ismail, 2019).

D. Rapid Application Development

Rapid Application Development (RAD) is an incremental software development model that is beneficial for short-term work. RAD is also one of the methods of developing an information system in a relatively short time. Developing a standard information system requires a minimum of 180 days, but using the RAD method can be completed in 30-90 days (Noertjahyana, 2002).

E. Blackbox Testing

Blackbox testing is a software quality test that tests the software's functionality. This test also aims to find incorrect functions, interface errors, data structure errors, initialization and termination errors, and performance error testing (Wijaya & Astuti, 2021).

RESEARCH METHODS

The method used in this research is Rapid Application Development (RAD) as shown in Table 1. RAD is an incremental software development process model, especially for short-term processing times (Sutinah et al., 2021). RAD is also a software process model emphasizing short-life circle development and an adapted version of Waterfall. RAD is structured into several stages that are interconnected and structured with each other. The stages in this method include Requirements Planning, User Design, Construction, and Cutover (Sukirman et al., 2023).

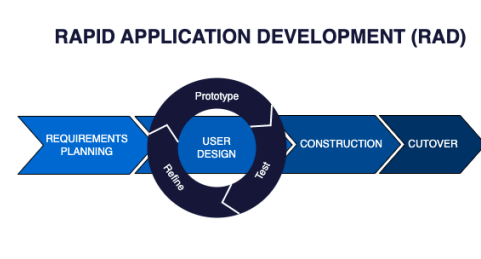


Figure 1. Rapid application development life circle

A. Requirements Planning

Requirements planning is the stage of identifying the objectives of system development, identifying needs, and collecting data or information related to research. Meanwhile, after identifying user needs, it is based on an actualization report containing ideas for efforts to solve issues related to the needs required by users, which consists of students and lecturers for the lecturers role requirements needs consists: (1)

Creating an RPS can be done by lecturers by selecting an option which will then display the data required by the RPS template such as courses, course codes, course credit weights, etc.; (2) RPS creation can be generated into a document file, namely PDF, automatically. For the student's role: (1) Students can see the RPS created and the Learning Plan for the courses in the semester the student will or has taken.

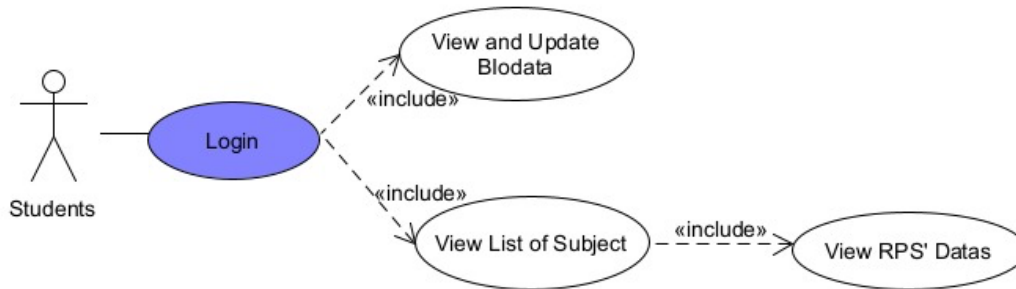


Figure 2. Student Use Case Diagram

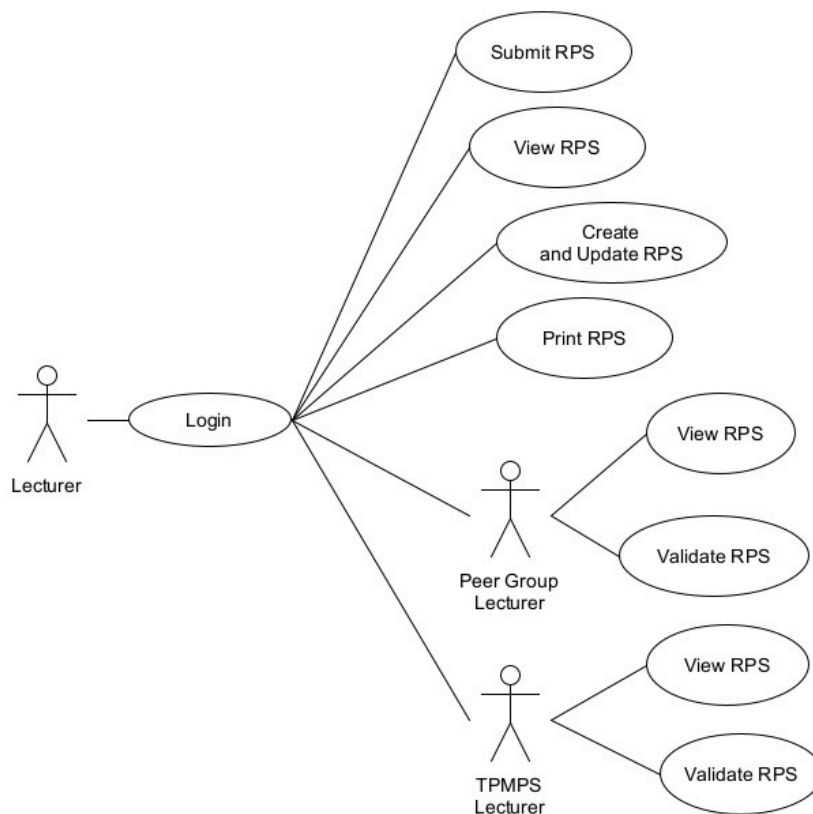


Figure 3. Lecturer use case diagram

B. User Design

Figure 2 shows that users with the student role must first log in to access the features

available on the system. Login can be done in 2 ways: using a username and password or using student SSO. After the user authenticates the

login, the user can access the features available on the system, including viewing and updating biodata and viewing RPS. Student use case definitions can be seen in Table 1.

Table 1. Student Use Case Diagram Definition

Use Case	Define
Login	At this stage, users with the student role can log in by entering their username and password. Students can also log in using SSO by entering their student SSO username and password.
View and Update Biodata	At this stage, students can see the student's profile biodata and update the data content.
View the Course List	In this case, students can see a list of RPS for each course in the study program.
Viewing RPS	At this use case stage, the user can see the RPS that was previously created

The lecturer uses the case diagram image above to show that users with this lecturer role are

Table 2. Lecturer Use Case Diagram Definition

Use Case	Define
Login	In this phase, users with the lecturer role need to log in by inputting their username and password to access the features contained in the system.
Viewing Profile	Use case Viewing biodata is the stage where the user can see the user profile biodata.
Viewing RPS	At this use case stage, the user can see the RPS that was previously created.
Create RPS	At this stage, users with the lecturer role can create RPS by inputting course choices, semesters, etc.
Update RPS	At this stage, the user can edit and update the RPS created by changing several attributes in the RPS itself.
Print out RPS (PDF)	At this stage, the user can generate the RPS that is being viewed and made into a document file.
Delete RPS	At this stage, the user can delete the existing RPS.
Signed RPS	In this use case, users with the lecturer role can upload signatures as required from the RPS content data.
Validate RPS	In this use case, users with the roles of peer group lecturers and TPMPS lecturers can validate the RPS

RESULTS AND DISCUSSION

The first stage of the RAD method is requirements planning. At this stage, the writer/researcher analyzes the needs and objectives of system development and collects data or information related to research. Next is the user design stage, which contains the flow or flow of the system and an overview of the initial concept of the system or mockup. The next stage is construction and implementation. At this stage, the implementation of the previous stage is carried out into a program code. And the final stage is cutover. At this stage, testing is carried out on the finished system. This testing is intended so that users or developers can

required to authenticate their login using their username and password first to access the features in the system. After that, the user enters the system with the features available on the system, such as viewing profiles, viewing RPS, creating new RPS, updating RPS, deleting RPS, and generating RPS into document files, and there are two sub-roles, namely lecturer as peer group and TPMPS which can validate RPS expectations. Table 2 shows the use case definitions.

C. Implementation

At the implementation stage, developers develop the system into a program. This stage will be structured into a framework for the work stages and writing source code for the system, integration units, and system testing. The developer and testing team themselves consist of the same entity to ensure the system runs smoothly and the results of this development can meet the expectations aimed at users.

determine whether the system being developed can be launched or released.

A. Requirements Planning

The Planning Requirements needed in developing the learning plan module for this semester are as follows: (1) The lecturer role can create an RPS by inputting the CPL, CPMK, bibliography that will be used in the course/RPS, the contents of the RPS itself such as subject matter, teaching materials, and required documents related to the RPS after the lecturer has finished inputting the RPS data according to the template provided. Lecturers can submit RPS, which will later be validated by the Lecturer's Peer group, the Study Program Quality

Assurance Team (TPMPS), and will be ratified by the Head of Study Program Role;(2) After the Head of the Study Program has ratified the RPS, students can see the list of RPS that have been ratified.

B. User Design

At this stage, the system requirements obtained at the Requirements planning stage are designed. The design will be a business process for the information system that will be developed.

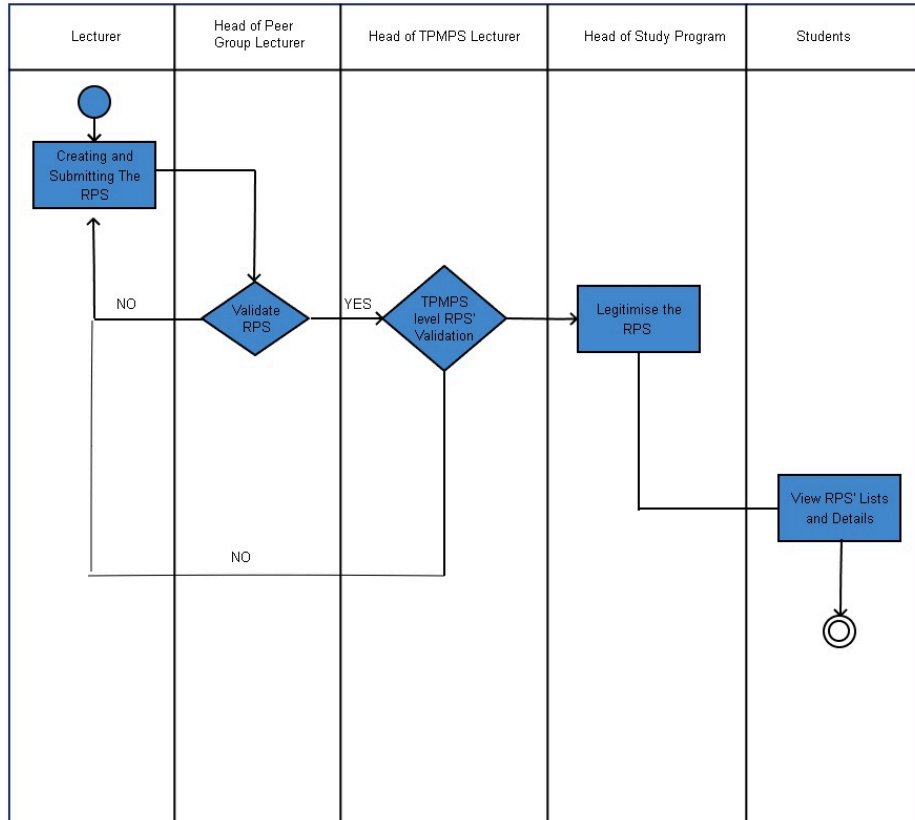


Figure 4. Business process of the RPS module

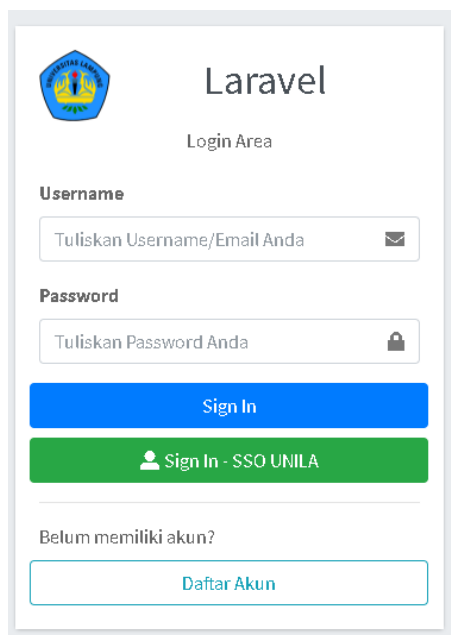


Figure 5. Login page interface

The picture above shows the flow of making the RPS and submitting, which the Lecturer carried out. Then, the proposed RPS will be validated by the Peer group lecturer. If the RPS submitted is valid, then the RPS will continue to be validated by the TPMPS lecturer. If the RPS submitted is invalid, the lecturer must create and submit the RPS again. Then, the RPS validated by the TPMPS lecturer will continue to the RPS Ratification stage, which the Head of the Study Program will carry out. This also happens like the previous stage; if the TPMPS lecturer disapproves/validates the proposed RPS, the stage will return to the beginning; lecturers must create and submit a new RPS. Next, there is the stage of validating the RPS. At this stage, the Head of the Study Program only acts to validate it because the Peer group Lecturers and TPMPS Lecturers have carried out sufficient checks. If the proposed RPS has been approved, students can see it.

C. Implementation

In Figure 5, the login page to the information system is displayed for all users, namely lecturers, study program heads, and students. This login display contains the username and password form of the user registered in the system. Then, there is a sign-in button to enter the system after filling out the username and password form. There is also a UNILA SSO sign-in button, which has also been integrated with UNILA SSO so that users can more easily log in directly to the system. In the picture, the login page to the information system is displayed for all users, namely, lecturers, heads of study programs, and students. This login

display contains the username and password form of the user registered in the system. Then, there is a sign-in button to enter the system after filling out the username and password form. There is also a UNILA SSO sign-in button, which has also been integrated with UNILA SSO so that users can more easily log in directly to the system.

Figure 6 displays the lecturer's teaching list; there is an input option to select the courses taught by the lecturer according to the specified period. Then, there is a table containing the course code, course name, class name, course credits, course type, and action buttons to show details of the selected course

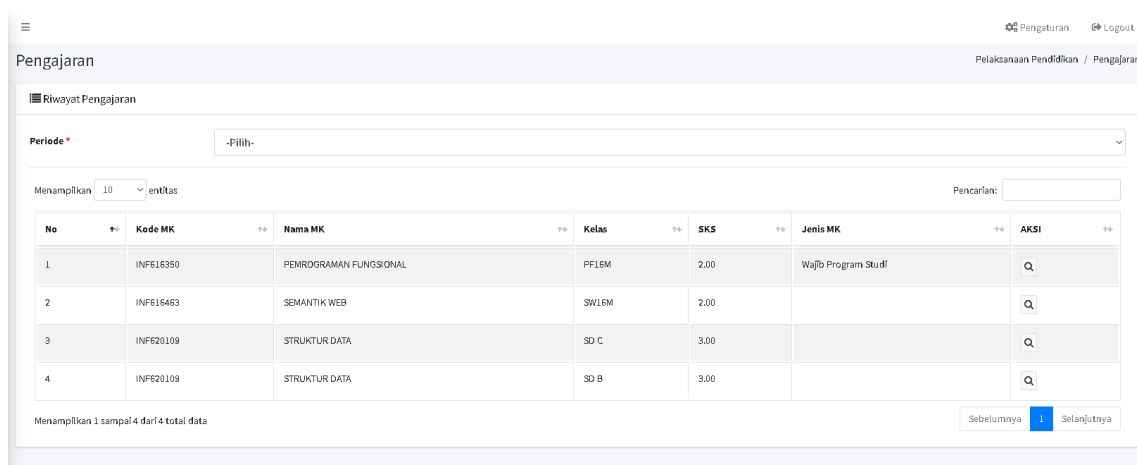


Figure 6. Interface of lecturer's teaching list

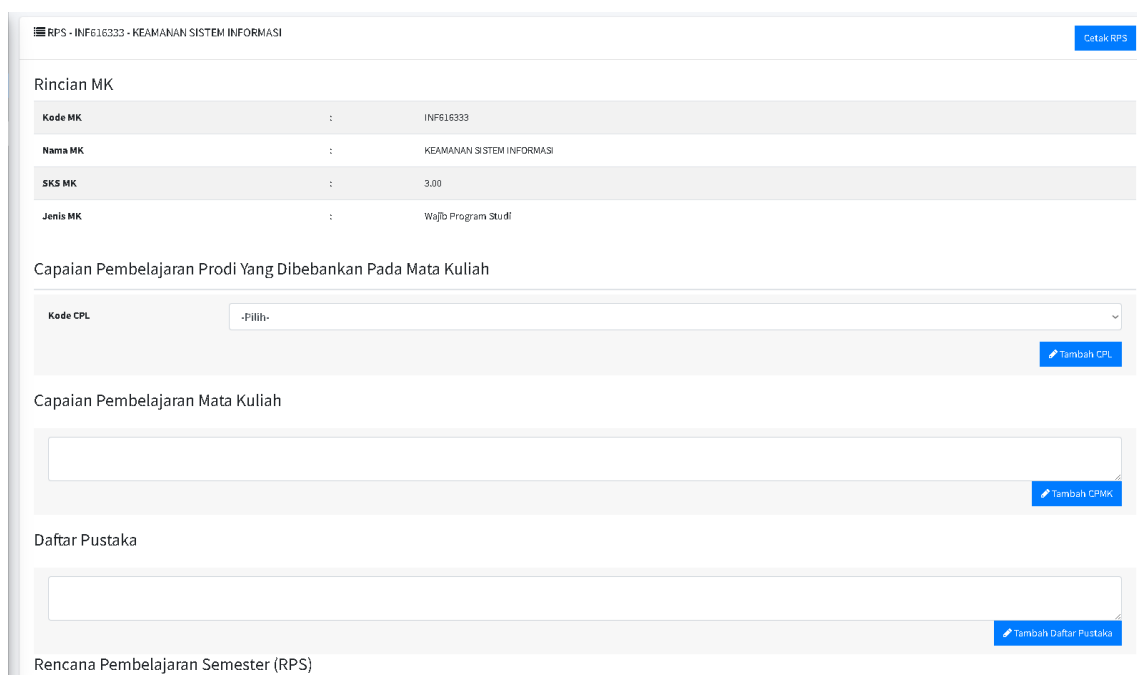


Figure 7. Detail of the lecturer's courses Interface

Figure 7 shows the lecturer's course detail display, such as course details containing code, name, credits, and course type. Also, on this course detail page, there is a table of study program learning outcomes (CPL), course learning achievements (CPMK), bibliography table, RPS table, and RPS submission history table. On the lecturer's course details page, there is a Print RPS button to generate course details and RPS details into a PDF file.

Figure 8 displays the new RPS submission page; a table of details of the RPS will

be submitted. This table contains the specific objectives, subject matter, references, sub-subjects, methods, media, assignment activities, and an action button to edit the RPS details according to the column specified. Chosen. Then, there is a form for choosing a learning method between the case-based method and project-based method, and there is a link that will direct the user to Google Docs, which contains examples of each learning method that the user will choose.

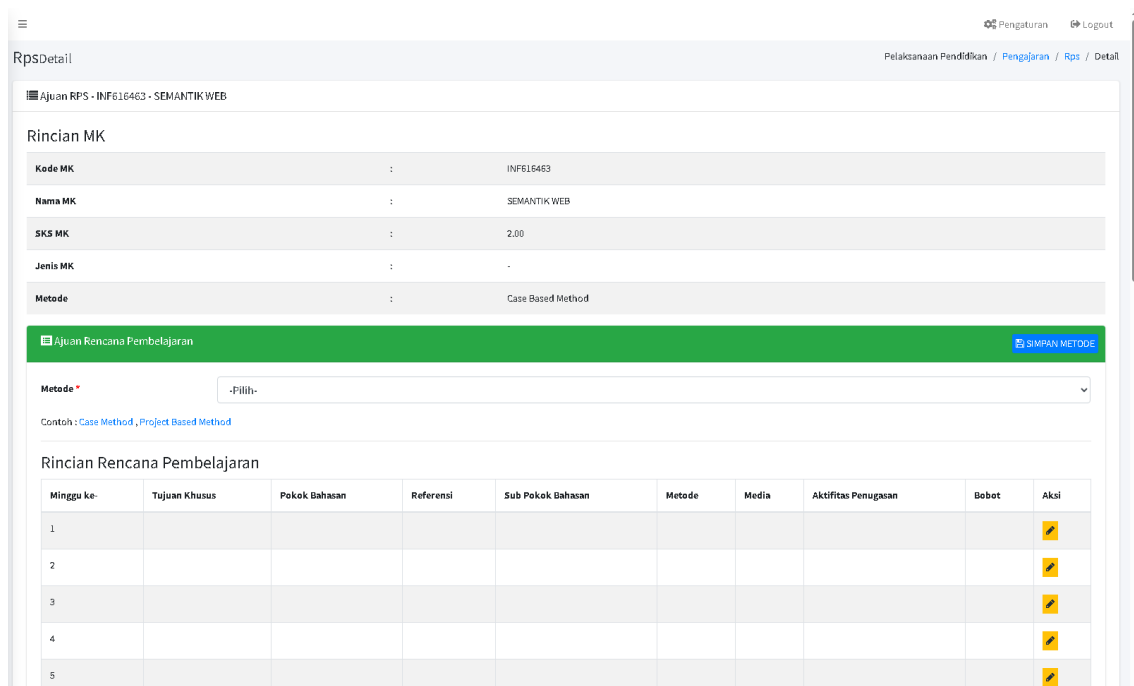


Figure 8. Interface of the new RPS submission page

Figure 9 depicts the RPS validation display for the peer group lecturer stage. There is a blue box that says, "You are in the peer group", which means that the lecturer who is entering the peer group lecturer validation page has access and has successfully accessed this page. Then there are options buttons: submitted, approved, rejected, and deferred. This option will bring up the RPS table according to the application status. In the picture above, in the submitted status, there is a table containing the RPS previously created and submitted by the lecturer. The submitted RPS table includes the name of the submitting lecturer, study program origin, course name, date the RPS was submitted, and an action button to validate the RPS.

Figure 10 contains a detailed display of the peer group lecturer's RPS validation stage, showing the contents of the previously submitted RPS list action button. This page contains data from the proposed RPS: course details, name of the submitting lecturer, study program learning outcomes table, course learning achievements table, bibliography table, RPS details table, and list of RPS supporting documents. Then, there is a form for inputting the inspection status and information by the peer group lecturer as a result of the validation that the peer group lecturer will carry out. Finally, a save validation button will save the validation form input along with its description and will continue the RPS validation to the TPMPS lecturer stage.

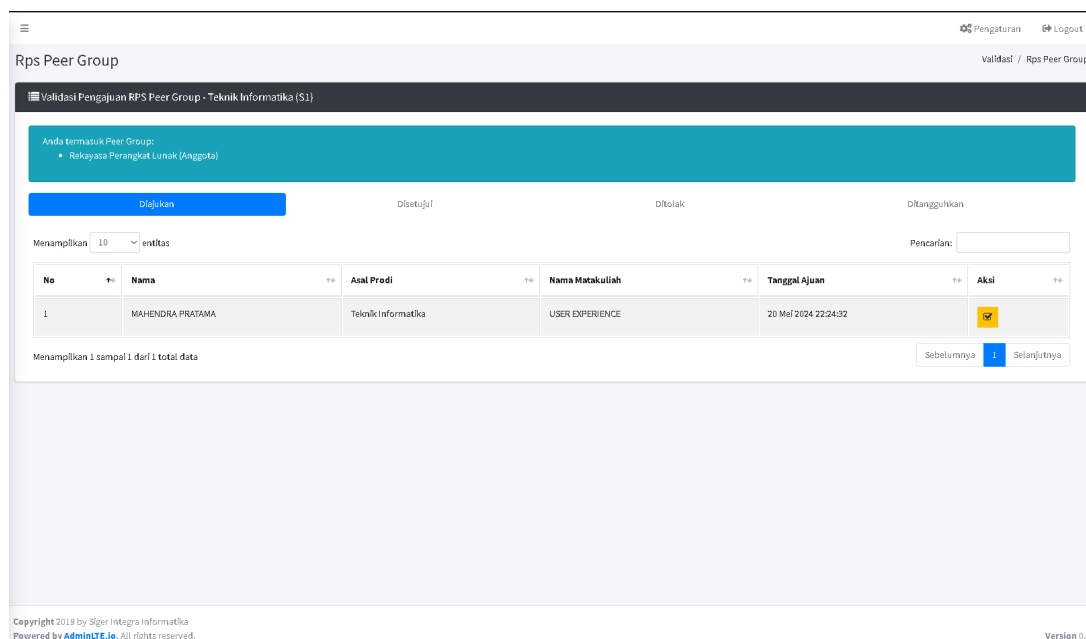


Figure 9. Interface of peer group lecturer stage RPS validation

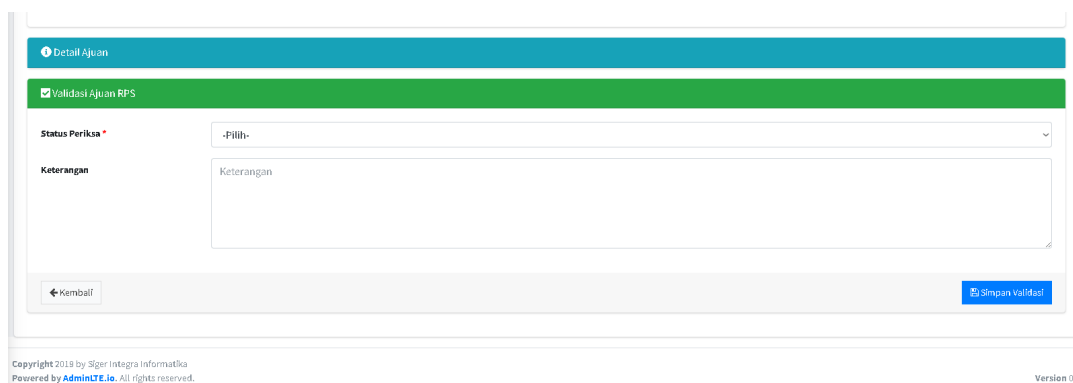


Figure 10. View details of RPS validation for peer group lecturer stage

Figure 11 displays the RPS validation list for the TPMPS lecturer stage; you can see a blue box that says “You are TPMPS,” meaning that the lecturer who entered this page is a TPMPS lecturer. Then, the TPMPS lecturer will be shown the application status of the RPS, from submitted, approved, rejected, and deferred. Then, there is also a list of RPS that have been submitted and previously validated by peer group lecturers in the form of a table containing the name of the submitting lecturer, the origin of the study program, the name of the course of the proposed RPS, the date of submission, as well as an action button to validate the RPS that is being submitted and has been validated by peer group lecturers.

Figure 12 shows the RPS validation stage for the TPMPS lecturer. This page will appear if

the TPMPS lecturer presses the RPS validation action button. On the RPS validation details page for the TPMPS lecturer stage, the data content of the proposed RPS is also displayed in the form of a course details table, study program learning outcomes table, course learning outcomes table, bibliography table, RPS details table, supporting documents for the proposed RPS. Details of the RPS application: There is a form for validating the RPS by selecting the application status for the RPS and the RPS validation information. Finally, there is a save validation button to save the results of the validation that the TPMPS lecturer has carried out, and this RPS validation stage will continue to the ratification of the RPS by the head of the study program.

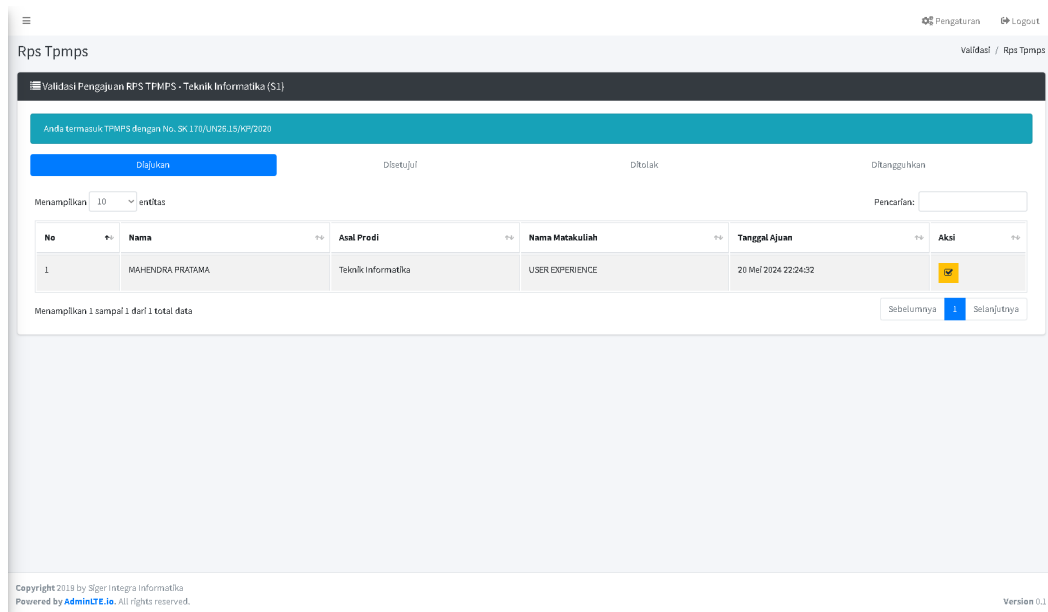


Figure 11. The interface of the RPS validation list for the TPMPS lecturer stage

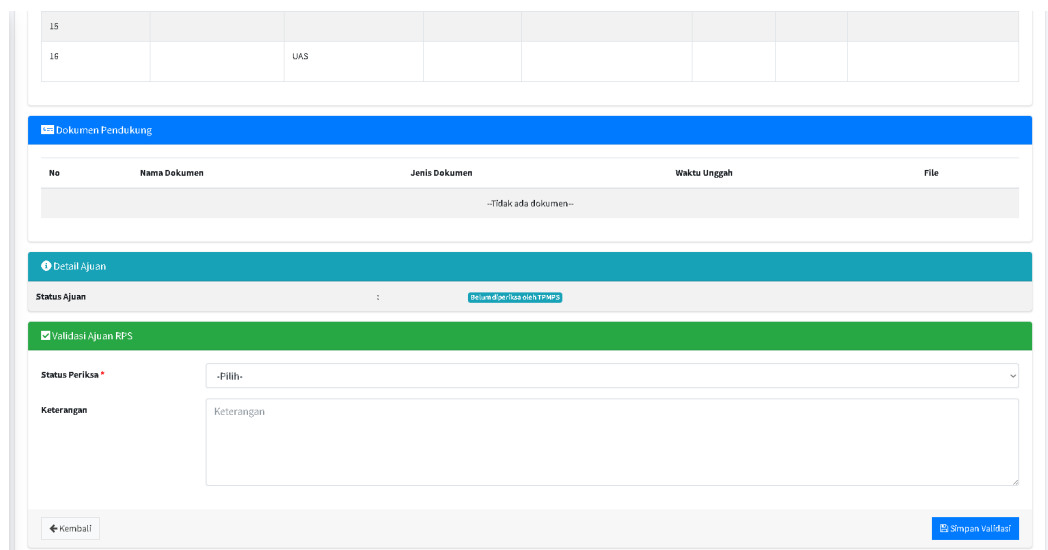


Figure 12. The interface of detailed RPS validation for TPMPS lecturer stage

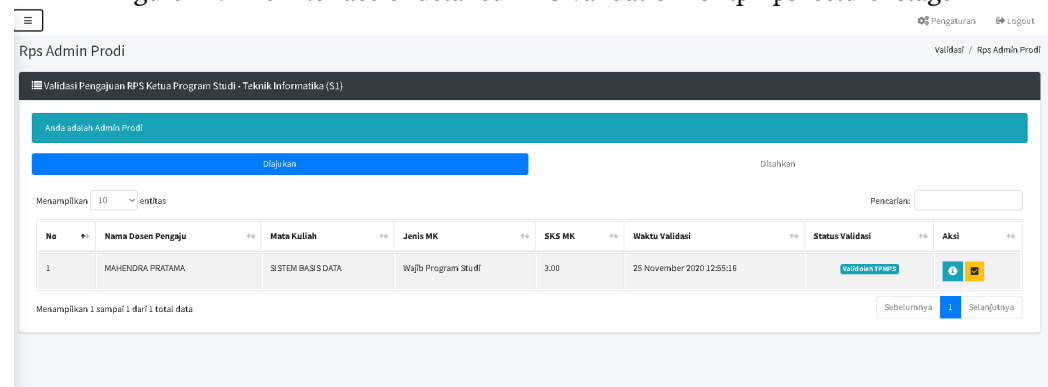


Figure 13. Interface of RPS approval list by study program chair

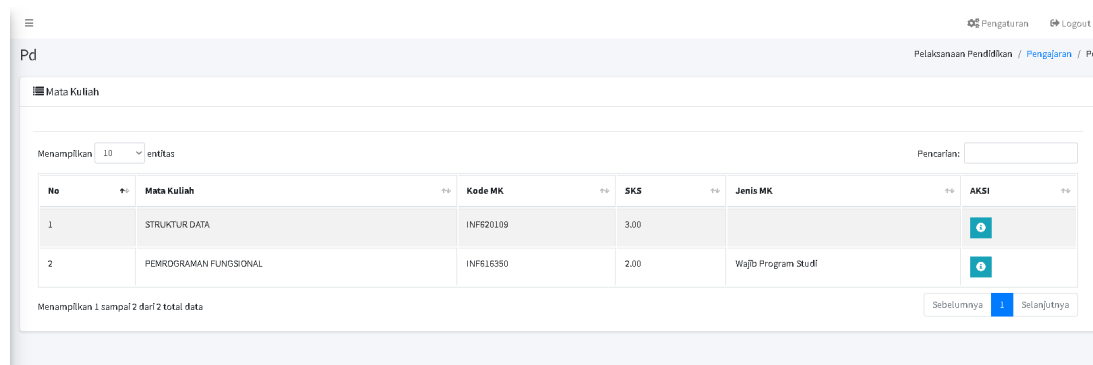


Figure 14. RPS list display in student role

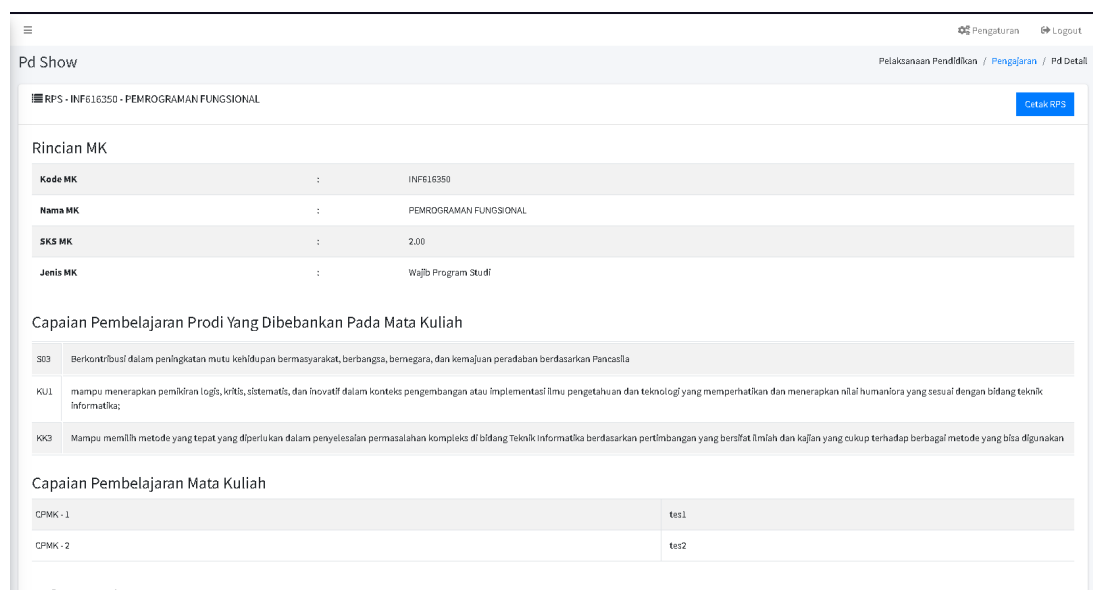


Figure 15. Interface of RPS details on student roles

Figure 13 depicts the RPS validation list by the study program head. Users with the role of study program head can access the RPS validation menu, and it will display a list of RPS that have been submitted and then validated by peer group lecturers and by TPMPS lecturers. This menu also has a choice of RPS application status, namely submitted and approved. In the proposed option, a list of RPS that have been submitted and have been validated by peer group lecturers and TPMPS lecturers will be in the form of a table with contents, name of the submitting lecturer, course, type of course, course credits, validation time, latest validation status, and an action button, namely RPS details, and the Validate RPS button. The RPS detail action button will display the details of the RPS, and the

Validate RPS button will change the status of the submitted RPS to a valid RPS and can be seen by students. In Figure 14, a list of RPS courses is approved by the head of the study program. This list is in the form of a table containing the course name, course code, course credits, course type, and an action button to view details of the course RPS. Figure 15 shows the detailed RPS display for the student role details of the RPS for the courses selected in the previous RPS list display. This detail contains details of the course, study program learning achievements, course learning achievements, bibliography, and details of the RPS, the data displayed is in the form of a table. On the student role RPS detail page, a print RPS button will generate the RPS into a PDF file in a preset template format.


			KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN UNIVERSITAS LAMPUNG FAKULTAS TEKNIK JURUSAN TEKNIK ELEKTRO PROGRAM STUDI TEKNIK INFORMATIKA		
Rencana Pembelajaran Semester					
Nama Matkul		Kode Matkul		SKS Matkul	
PEMROGRAMAN FUNGSIONAL		INF616350		2.00	
Otorisasi			Penyusun RPS		
			MAHENDRA PRATAMA		
CPL-PRODI (Capaian Pembelajaran Lulusan Program Studi) Yang Dibebankan Pada Mata Kuliah		Deskripsi CPL			
S03		Berkontribusi dalam peningkatan mutu kehidupan bermasyarakat, berbangsa, bernegara, dan kemajuan peradaban berdasarkan Pancasila			
KU1		mampu menerapkan pemikiran logis, kritis, sistematis, dan inovatif dalam konteks pengembangan atau implementasi ilmu pengetahuan dan teknologi yang memperhatikan dan menerapkan nilai humaniora yang sesuai dengan bidang teknik informatika;			
KK3		Mampu memilih metode yang tepat yang diperlukan dalam penyelesaian permasalahan kompleks di bidang Teknik Informatika berdasarkan pertimbangan yang bersifat ilmiah dan kajian yang cukup terhadap berbagai metode yang bisa digunakan			
CPMK (Capaian Pembelajaran Mata Kuliah)		Deskripsi CPMK			

Figure 16. Interface of RPS print

Table 3. Lecturer Use Case Diagram Definition

Test Class	Test Items	Types of Testing	Result
Checking	View the List and Details of RPS for Courses Taught by Lecturers	Blackbox	OK
	View RPS application history	Blackbox	OK
	Generate RPS	Blackbox	OK
	View the List and Details of Course RPS that the Head of the Study Program has approved in the student role	Blackbox	OK
Data Entry	Filling in RPS application data	Blackbox	OK
	Mengajukan RPS	Blackbox	OK

Figure 16 shows the RPS data displayed, namely, course name, course code, course credits, RPS applicant, study program learning achievements, course learning achievements, bibliography, lecturer in charge of the course, and RPS details table. This pdf file has also been arranged so that the format of each RPS will be the same as the University of Lampung logo, etc., which will also appear on the RPS printed page.

D. Cutover

Based on blackbox testing, testing on lecturer role And student role overall for each function the role of the lecturer and the role of student, the results were obtained testing average

results on average good results without any obstacles when testing with black box testing.

CONCLUSION

The conclusions obtained from the research that has been carried out are as follows: (1) The RPS module has been successfully developed based on the features that have been created, namely, submitting RPS, validating RPS, viewing RPS, and printing RPS by generating RPS into a PDF file; (2) Based on the results of testing the semester learning plan module with each one of the features such as creating and submitting the RPS, validation approval phase by the peer group lecturers etc., generating the RPS to a PDF file, using the black

box testing method, it was found that the test results were in accordance with the expected results and went well; (3) Based on the module developed in this research, this study program's portal system still does not have a menu for

adding data on Study Program Learning Achievements, synchronizing course data, and lecturer teaching classes.

REFERENCES

- Girum, T., Lentiro, K., Geremew, M., Migora, B., & Shewamare, S. (2020). Global strategies and effectiveness for COVID-19 prevention through contact tracing, screening, quarantine, and isolation: a systematic review. *Tropical Medicine and Health*, 48(1), 91. <https://doi.org/10.1186/s41182-020-00285-w>
- Mendikbud. (2020). *Keputusan Menteri Pendidikan dan Kebudayaan Republik Indonesia Tentang Indikator Kinerja Utama Perguruan Tinggi Negeri dan Lembaga Layanan Pendidikan Tinggi Di Lingkungan Kementerian Pendidikan dan Kebudayaan* (pp. 1–30)<https://dikti.kemdikbud.go.id/wpcontent/uploads/2020/11/Salinan-Kepmen-754.P.2020.pdf>
- Pradipta, R. A., Pratama, M., Muhammad, M. A., & Muda, M. A. (2021). Perancangan Sistem Portal Program Studi Dengan Menggunakan Framework Laravel Dan Database Postgresql. *Electrician*, 15(2), 134–141. <https://doi.org/10.23960/elc.v15n2.2199>
- Sukirman, Zaenuri, M., & Hasanah, L. Q. (2023). Pengembangan Rencana Pembelajaran Semester (RPS) Mata Kuliah Mahārat al-Istimā' Berbasis Kecakapan Abad 21. *Al-Ma'Rifah*, 20(1), 1–20. <https://doi.org/10.21009/almakrifah.20.01.01>
- Marthasari, G. I., Aminudin, A., & Munarko, Y. (2012). Implementasi Web Service untuk Mendukung Interoperabilitas pada Aplikasi E-Commerce. *Jurnal Teknik Informatika, Universitas Muhammadiyah Malang*, 1(1), 1–10.
- Prasetyo, D. A. B. (2022). Implementasi Information Schema Database pada PostgreSQL untuk Pembuatan Tabel Informasi dengan Menggunakan Python di PT XYZ. *JATISI (Jurnal Teknik Informatika dan Sistem Informasi)*, 9(3), 1961–1972. <https://doi.org/10.35957/jatisi.v9i3.2221>
- Ismail. (2019). Pemanfaatan Framework Laravel untuk Pengembangan Sistem Informasi Toko Online di Toko New Trend Baturetno. *Jurnal Skripsi (J. Scr.)*, 7(2), 232–238.
- Noertjahyana, A. (2002). Studi analisis Rapid Application Development sebagai salah satu alternatif metode pengembangan perangkat lunak. *Jurnal Informatika (J. Inform.)*, 3(2), 64–68.
- Wijaya, Y. D., & Astuti, M. W. (2021). Pengujian blackbox sistem informasi penilaian kinerja karyawan PT INKA (Persero) berbasis equivalence partitions. *Jurnal Digital Teknologi Informasi (J. Digit. Teknol. Inf.)*, 4(1), 22. <https://doi.org/10.32502/digital.v4i1.3163>
- Sutinah, E., Alfarobi, I., & Setiawan, A. (2021). Metode Rapid Application Development Dalam Pembuatan Sistem Informasi Pemenuhan SDM pada Perusahaan Outsourcing. *InfoTekJar: Jurnal Nasional Informatika Dan Teknologi Jaringan*, 5(2), 246–253. <https://doi.org/https://doi.org/10.30743/infotekjar.v5i2.3528>