ISSN : 1978-8282, Online ISSN: 2655-4275

Analysis and Design of Graduate Learning Outcomes (CPL) Assessment System

Prya Artha Widjaja[*1], Nico Abel Laia[2], Verio Aberossy Renedominick[3]
[1,2,3] Department of Informatics Engineering, Faculty of Natural, Computer and Mathematical Sciences, Matana University, Indonesia
E-mail: [*1]**prya.artha@matanauniversity.ac.id**, [2]nico.laia@student.matanauniversity.ac.id, [3]verio.renedominick@student.matanauniversity.ac.id

Abstract

To be competitive in this era, one of the essential needs of a university is good academic performance. How can a university determine that the academic performance of its curriculum is good? One of the methods that can be used is to analyze the Graduate Learning Outcomes (CPL). This assessment can be conducted to see whether the existing courses or curriculum are good or still need improvement. In practice at Matana University, this is quite difficult to do manually. There are many errors and misunderstandings among lecturers in conducting CPL assessments. This is compounded by the arrival of new lecturers who do not understand how to analyze CPL, requiring retraining every semester. To address this, a CPL calculation analysis system was designed. This system was developed using the prototype method and resulted in a system prototype. This system will integrate course learning plan, graduate learning outcome and course learning outcome. The design will use UML, and ERD is used for the database design. The next step is to create the system and implement it.

Keywords *— Graduate Learning Outcomes, Analysis and Design*

1. INTRODUCTION

The heart of education is curriculum and to achieve the established curriculum, appropriate learning outcomes are needed [1]. Graduate Learning Outcomes (CPL) or in short Learning Outcomes (LO) are needed to assess each course and curriculum is already appropriate or not. It is not easy to assess the student learning outcomes [2], [3]. To determine whether the provided Learning Outcomes meet the standards, calculations are necessary [4].Every lecturer should assess all the assignments and exams to calculate the value of CPL. Each component should be available in Course Learning Plan (Rencana Pembelajaran Semester - RPS) [5]. Currently at Matana University, every lecturer should do the calculations manually. These are tedious works, some lecturers need longer time to calculate the score, and other lecturers make a mistake because of a lack of understanding to calculate the CPL. If there are new lecturers, usually lacking knowledge to calculate the CPL, therefore require training again. To solve this, we want to create a system that can be used by every lecture to calculate the learning outcomes easily. There are some researchers trying to solve this problem using spreadsheets [6]. But this approach still limits the usefulness of the system. We want to create an assessment system that every lecturer can start from creating Course Learning Plan (RPS) and finally will calculate the CPL for each course automatically. Some researchers already create a system like this and give a good result [7], [8]. Researchers try to create a system that can be used by lecturers, heads of department and dean of the faculty. The application will

integrate RPS, CPL and Course Learning Outcome (CPMK). This system will be created using NextJs, but for now we will discuss the analysis and design of the system and create the prototype.

2. RESEARCH METHOD

2.1. Data Collection

First thing we must do is collect the data to start analysis and design the system. Because we are using the prototype model, so data was collected step by step according to the cycle. We collected the data by conducting interview with lecturers, head of study program and head of academic. Beside interview, we also using observation while doing assessment manually.

2.2. Prototype Model

Although some researchers found that using agile method like Extreme Programming useful for developing web-based application [9], [10]. The software development model that we used is a prototype model. We are choosing this model because the client has general objectives for the application but does not give detailed features, also we need the application's user interface to be very simple and easy to use. Prototype model may offer the best solutions for these situations [11]. There are many types of prototyping like throwaway prototyping and evolutionary prototyping. This research using evolutionary prototyping because some researchers already used it and feel this method is appropriate for developing a software [12].

2.3. Analysis and Design

Because this application will be developed using object oriented programming languages, analysis and design will be using Unified Modeling Language (UML) [13]. We will be using Use Case diagram to show the relations between functional requirements of the systems and actors (external objects). Activity diagrams will be used to show the flow of the system. We also use class diagram to show relation each class

2.4. Database

All data will be saved in database, and we choose to use relational database. Although the database relation can be shown using class diagram, we choose to create Entity Relationship Diagram (ERD) to show the relation in database.

3. RESEARCH RESULTS AND DISCUSSION

The first thing we did was collect data from head of academic to understand the flow of the system that we want to create. From the head of academic, we found that the system for head of study program should design first. After interviewing the head of study program, the

information we get is that the head of study program should enter courses based on each curriculum, each course should have a lecturer assign to it. The head of study program also must input all Graduate Learning Outcomes (CPL) for his/her study program and assign CPL needed for each course. Finally, the head of study program can assess CPL for his/her study program. Use case diagram for head of study program can be seen in Figure 1.

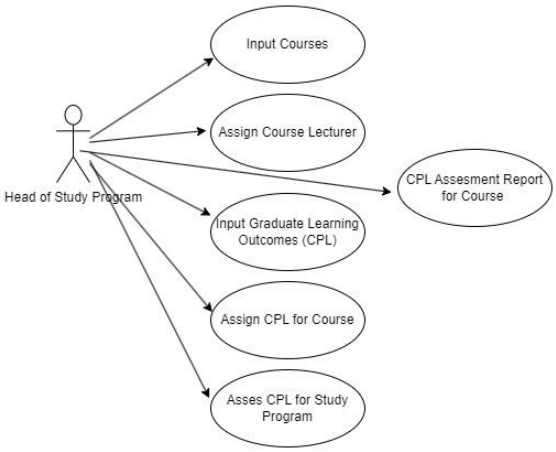


Figure 1. Use Case Head of Study Program

The head of study program should input all the courses in their study program into the application according to their curriculum. Because the curriculum could change every 4 – 5 years, the course in each curriculum could be different. The next step is input the lecturer and assign each lecturer into course. This lecturer will become course lecturers and will create an Course Learning Program (RPS) and do CPL assessment for their course. Next task for the head of study program is to input all the Graduate Learning Outcomes (CPL) for their study program. After all CPL have been inserted, the head of study program should assign every CPL appropriate for each course. Each course at least has one CPL and there are no limitations but usually at most has five CPL. Finally, the head of study program can has an assessment of CPL for their study program and see the report.

After all the flow for head of study program finish, we start to create the web design for the head of study program, starting from dashboard as you can see in Figure 2. Other features we can see in Figure 3, like how to add and change information about courses, also a report that can show all the courses that pass the CPL assessment. After that we create a user flow that shows every step of the web page as you can see in Figure 4.

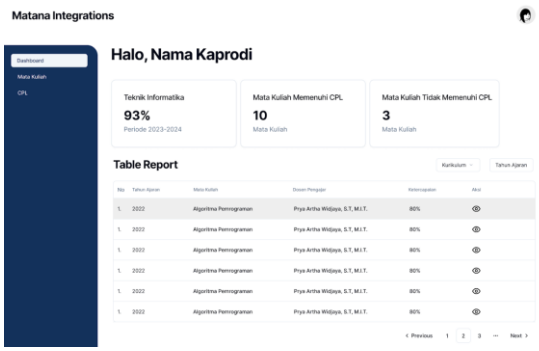


Figure 2. Dashboard for Head of Study Program

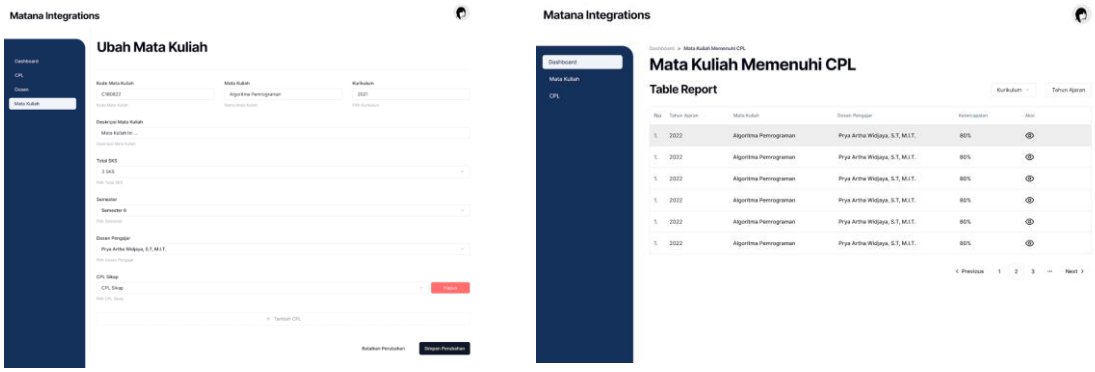


Figure 3. Features for Head of Study Program

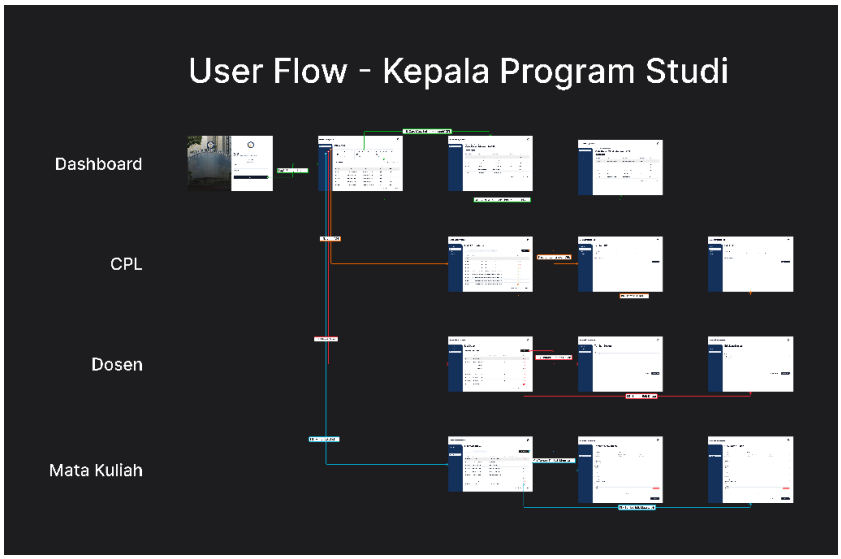


Figure 4. User Flow for The Head of Study Program

After the design prototype for head of study program finish, the next cycle is to collect information and design the system for course lecturer. The course lecturer should able to create course learning plan (RPS), create course learning outcomes (CPMK) and sub-course learning outcomes (sub-CPMK), assign the assessment weight and do the assessment for graduate learning outcomes (CPL). The use case is in Figure 5.

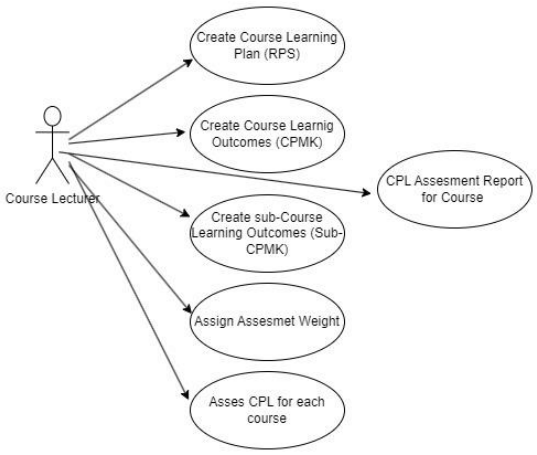


Figure 5. Use Case for Course Lecturer

Each course lecturer can access their course if already been assigned by their head of study program, if has not been assigned then they cannot access the course. As a course lecturer, the first thing to do is create a course learning plan (RPS) for each course there have been assigned. In every course they can see the graduate learning outcome (CPL) that has been assigned by the head of study program for the course. While creating an RPS, the course lecturer will create Course Learning Outcomes (CPMK) based on the graduate learning outcomes (CPL) that has been assigned. At least one CPMK must be based on one CPL. After the course learning outcomes has been defined, the course lecturer should create sub-course learning outcomes (sub-SPMK) based on course learning outcomes (CPMK). Every course lecturer must assign assessment weight for every assignment and exam in its course. These assignment weight should be based on CPMK so we can do an assessment for the Graduate Learning Outcomes (CPL) for each course. To do an assessment for each course, the lecturer must put all the grades for each assignment and exams, the system will calculate the grade with the assessment weight and show the result.

The view of the course lecturer is like the head of study program. First is a Dashboard, where the course lecturer can see the list of course his/her teach, the summary of courses that pass the assessment and the average grade of CPL assessment. Dashboard view in Figure 6.

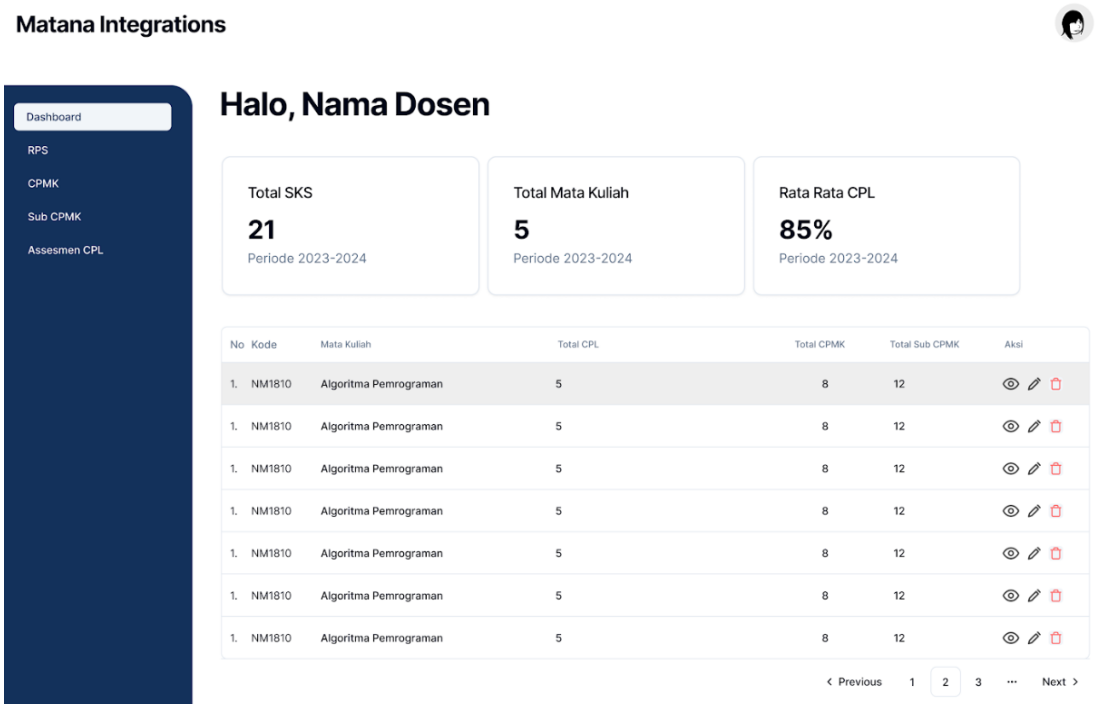


Figure 6. Dashboard for Course Lecturer

The course lecture can create RPS as we can see in Figure 7, also can add CPMK (Figure 8) and sub-CPMK (Figure 9). Finally, the course lecture can assign the assessment weight and do CPL assessment as we can see in Figure 10. User flow the application for course lecturer can be seen in Figure 11.

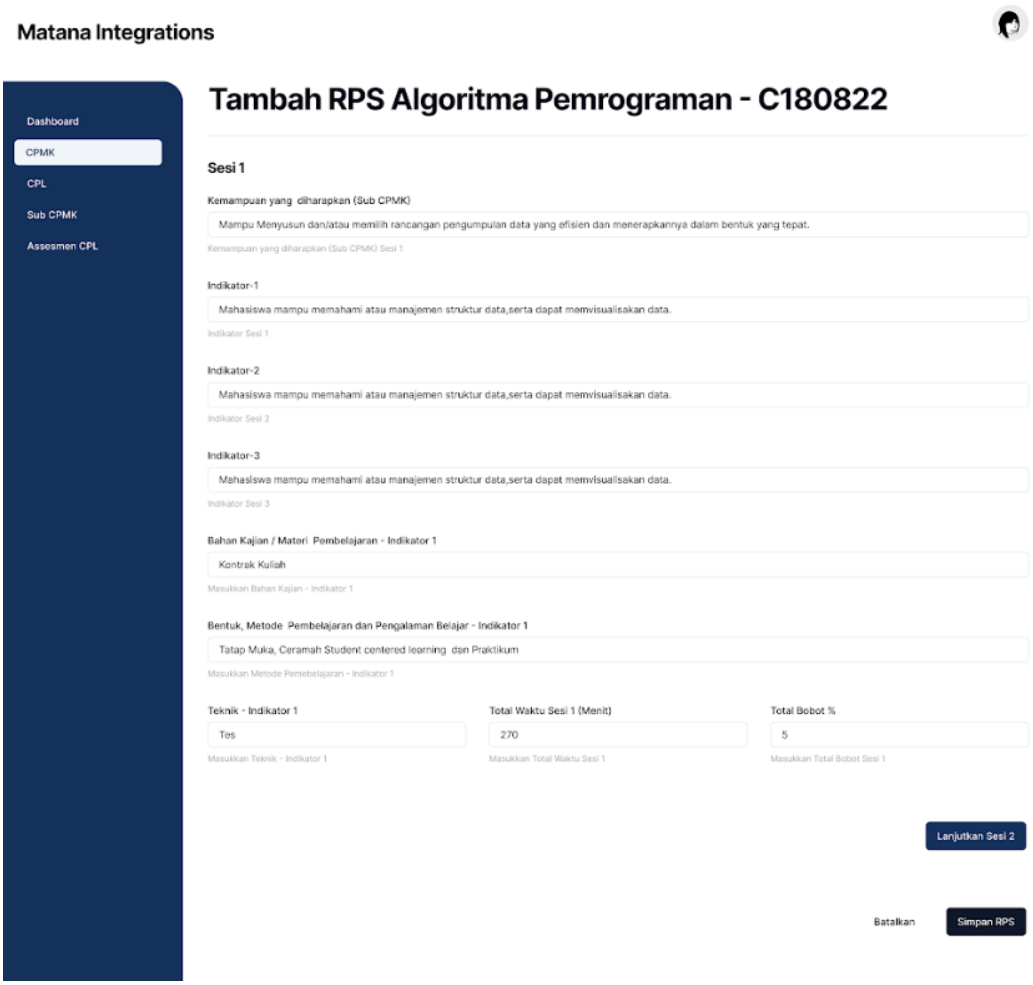


Figure 7. Create Course Learning Plan for Course Lecturer

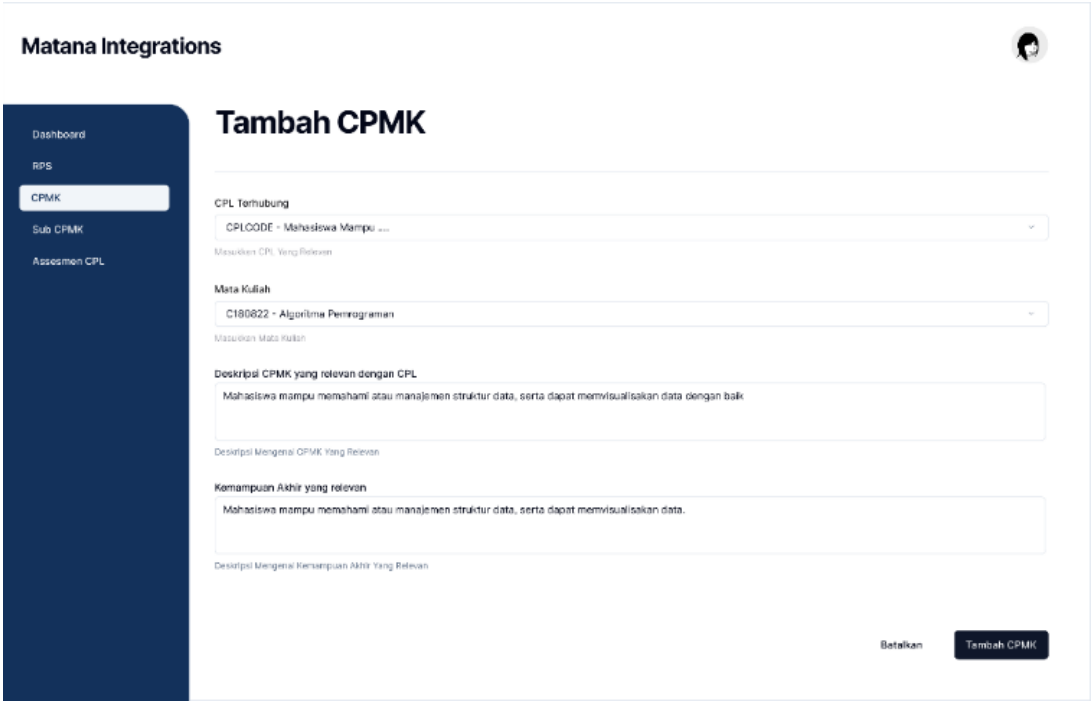


Figure 8. Add Course Learning Outcomes (CPMK)

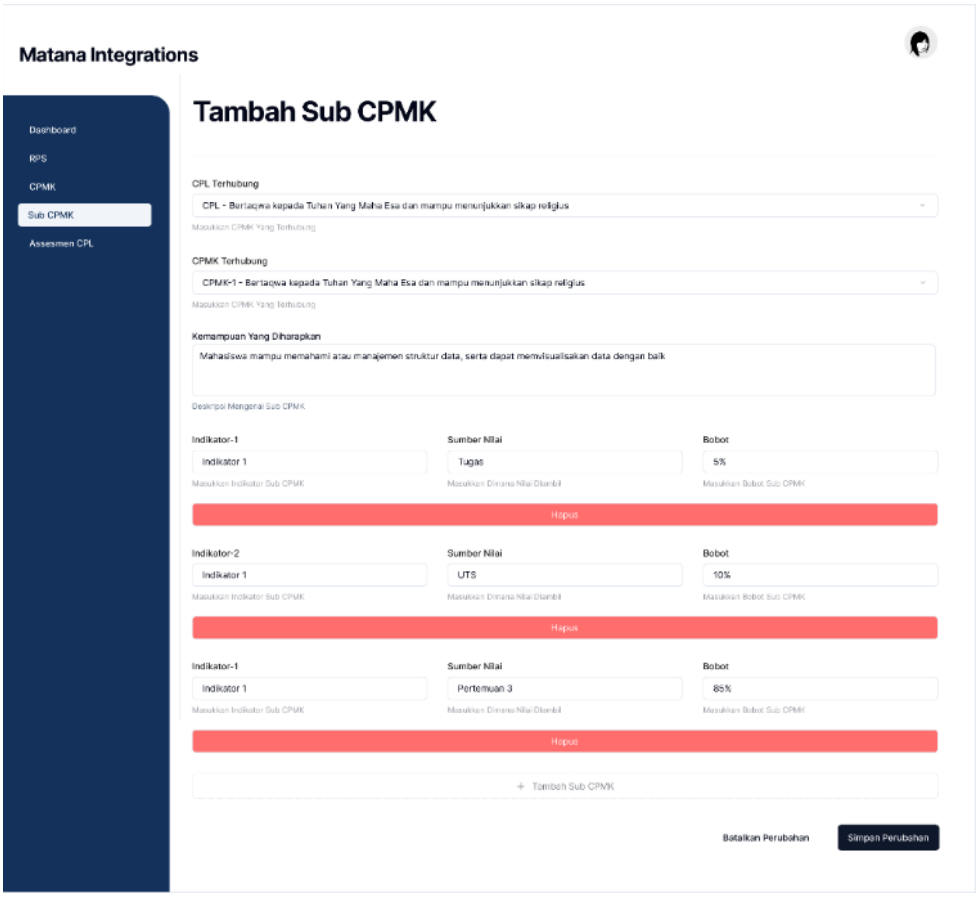


Figure 9. Add sub-Course Learning Outcomes (Sub-CPMK)

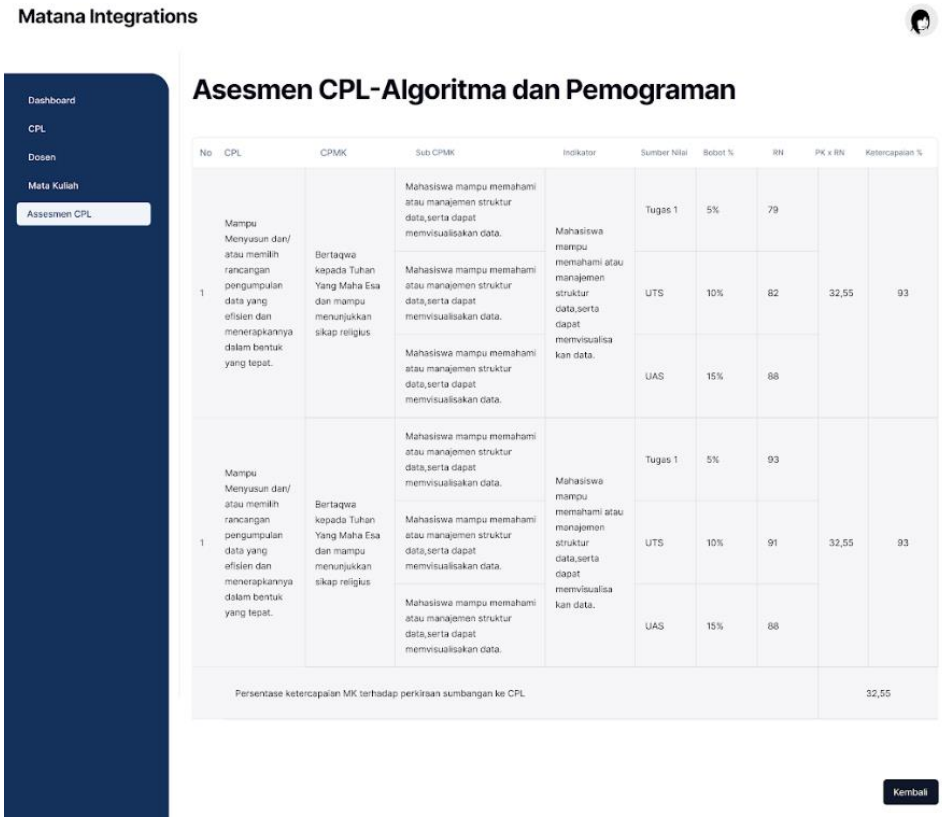


Figure **10.** Assessment of Graduate Learning Outcomes (CPL)

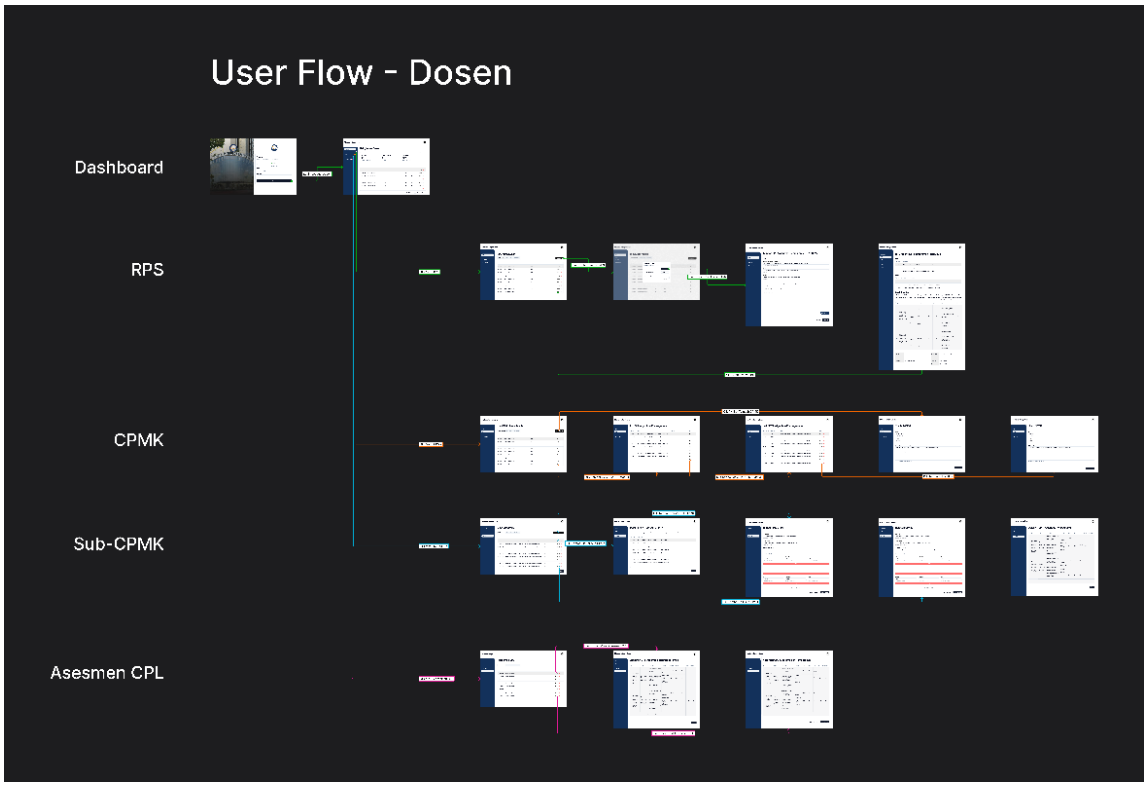


Figure 11. User Flow for Course Lecturer

After finishing the course lecturer cycle, the next cycle in this prototype method is to create an application for the Dean. The Dean is straight forward, here the dean can see the result of CPL assessment for every program study in its faculty.

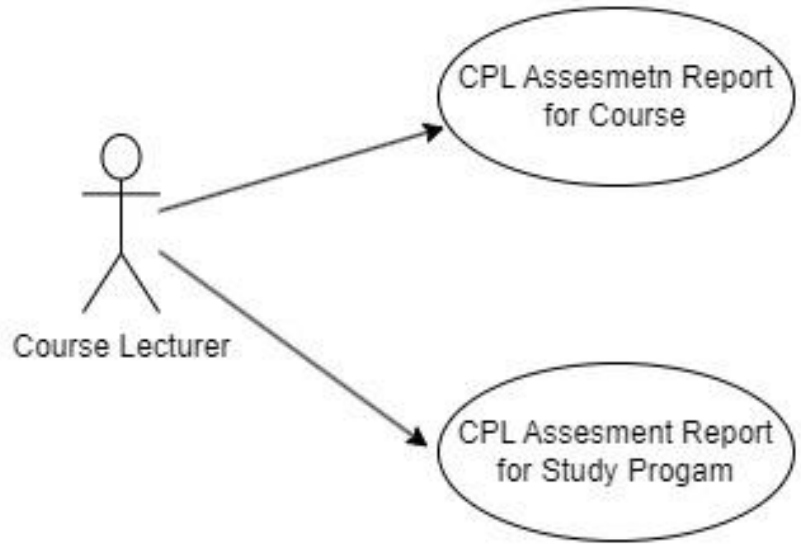


Figure 12. Use Case for Dean

In Dashboard, the dean can see all the study program in its faculty and the grade of CPL assessment for each study program. Dean also can see how many courses achieve the standard value that has been set and how many courses get CPL grade that below the standard. Dashboard can be seen in Figure 13 and the user flow of the application can be seen in Figure 14.

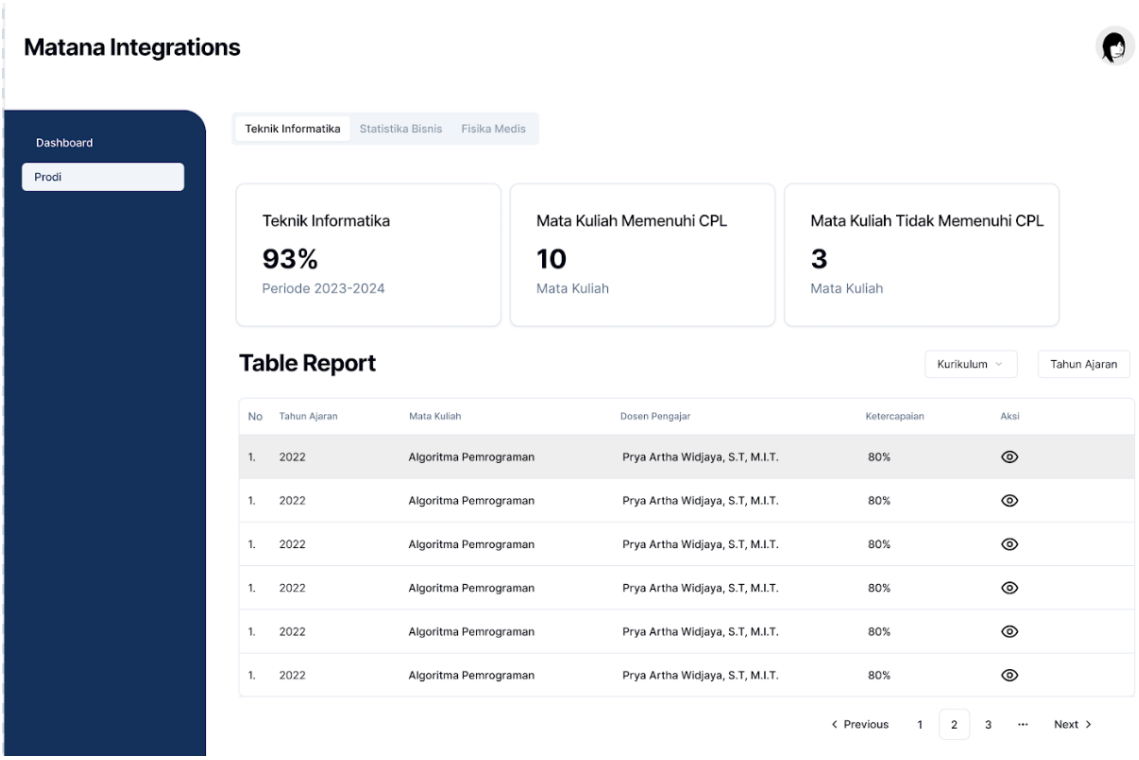


Figure 13. Dean's Dashboard

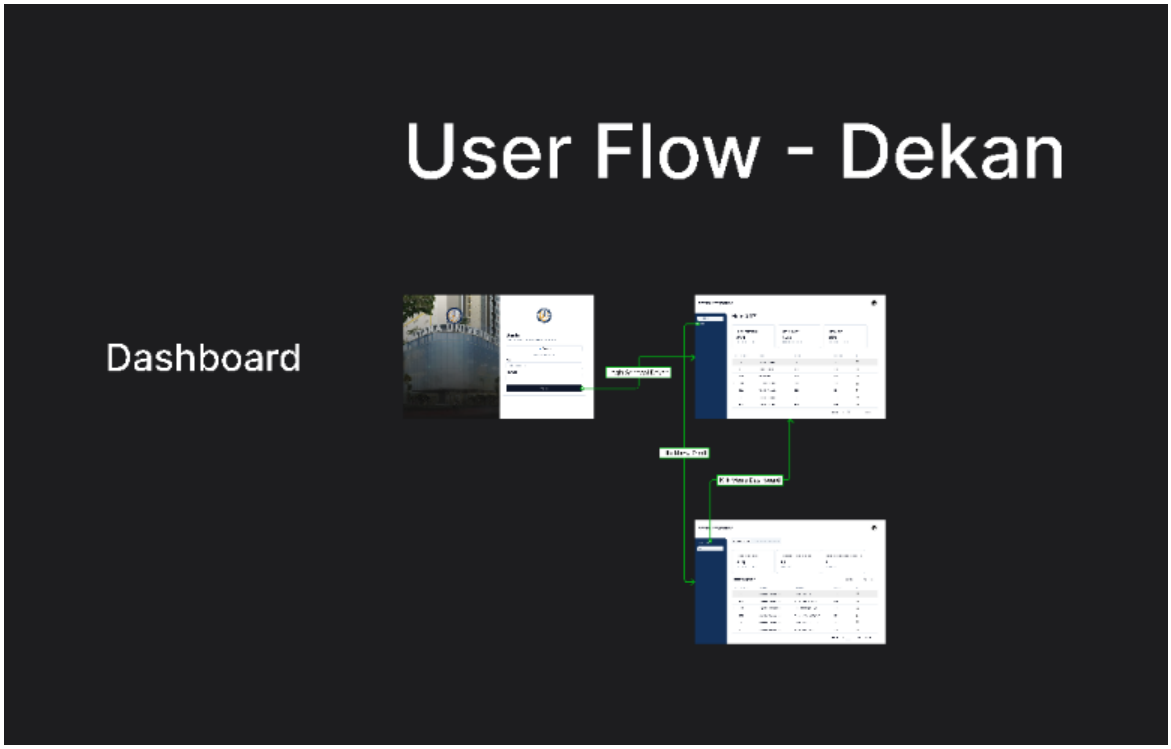


Figure 14. User Flow for Dean

This cycle finishes all the requirements and the prototype for the application. The next step is to design the activity diagram as you can see in Figure 15.

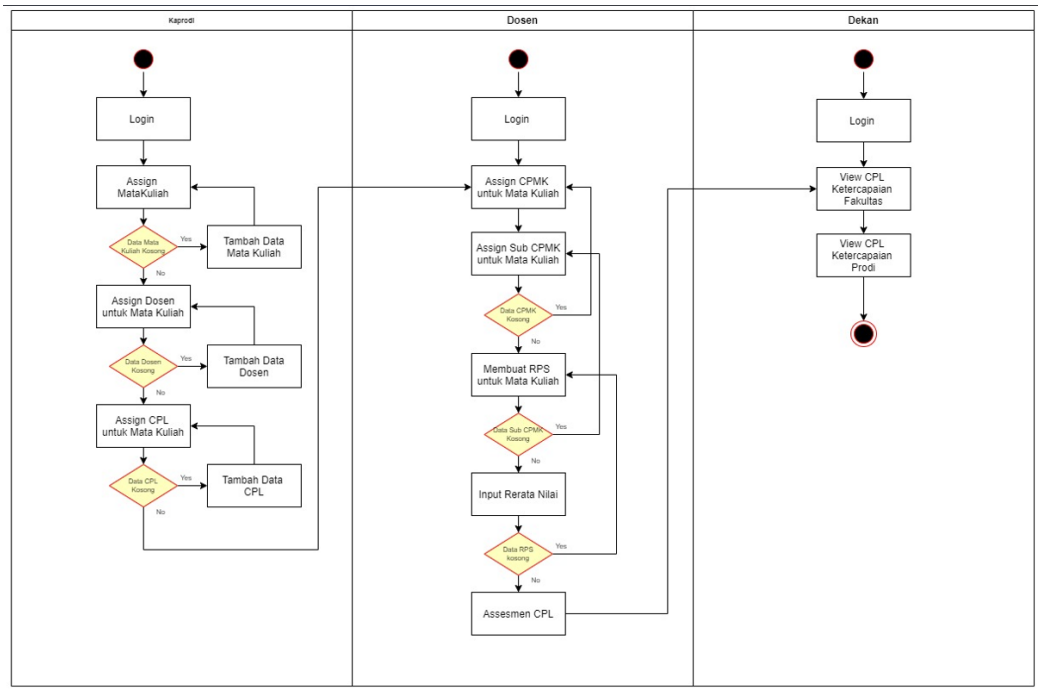


Figure 15. Activity Diagram for CPL Assessment System

After the activity diagram, the next step is to design the class and relationship between each class. This is needed for the next step, that is to create the application an implement it. The class diagram can be seen in Figure 16.

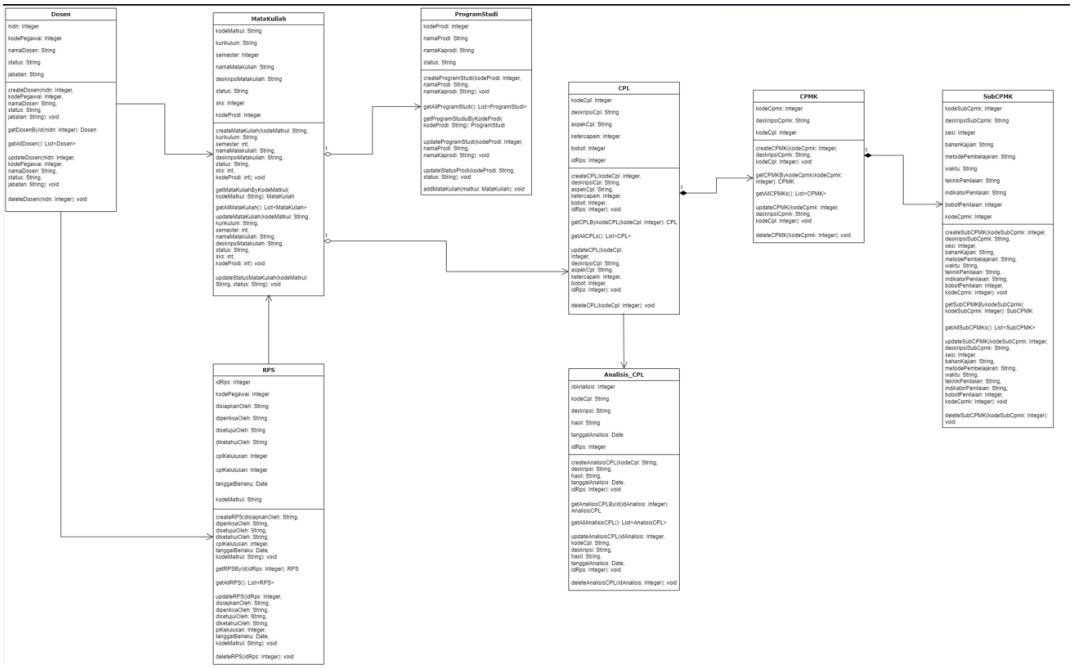


Figure 16. Class Diagram for CPL Assessment System

While designing this system, we were considering using a relational database. Why we choose relational database and not no sql database, simply because it is proven and the data is still manageable. To help the creation of the system, we create an Entity Relationship Diagram (ERD) that can be seen in Figure 17.

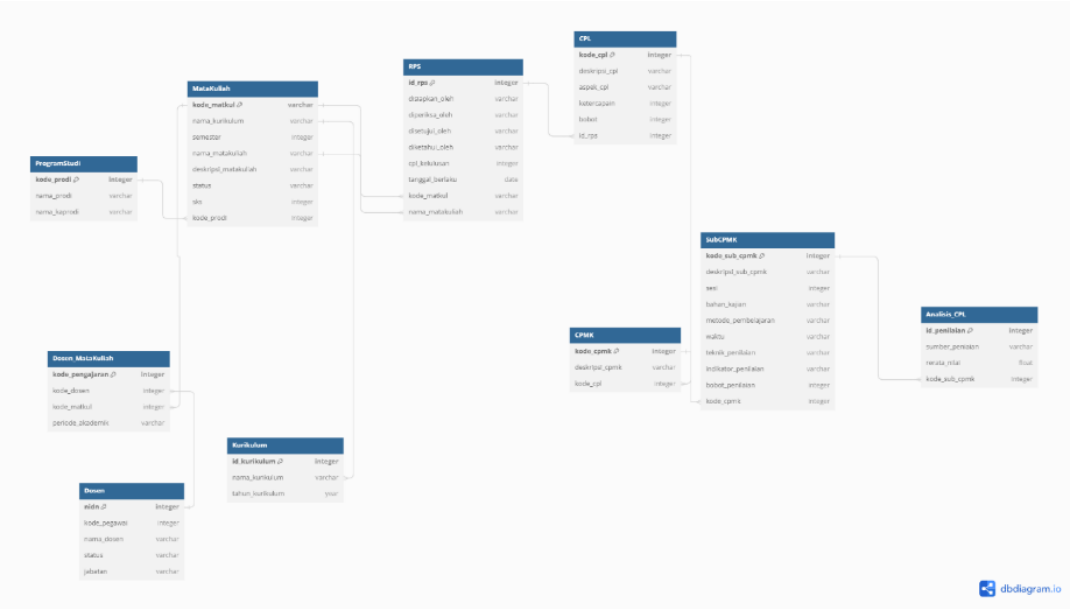


Figure 17. ERD for CPL Assessment System

There will be ten tables to create. Those tables are ProgramStudi, MataKuliah, RPS, CPL. CPMK, subCPMK, Analisis_CPL, Dosen, Dosen_MataKuliah and Kurikulum.

4. CONCLUSION

This analysis already covered most of the requirements needed to create a Graduate Learting Outcomes (CPL) Assessment system. From the analysis, we already created the design and built prototype from it. We already know the flow of the system, head of study program should input all the graduate learning outcomes (CPL) into every courses first, after that the course lecturer can input course learning outcomes (CPMK) and sub-course learning outcomes (sub-CPMK). The course lecturer also must put the assessment weight for each assignment and exam so the CPL can be assessed. Finally, the course lecturer, head of study program and dean can view the result of assessment for each course. The next step that should be taken is to create the application and implement it.

5. SUGGESTED

This research only covers one faculty with multiple study program, next research should include more than one faculty with multiple study program.

6. REFERENCES

[1] T. A. Wahyu, L. Lufthansa, and P. P. Setiani, “Analisis Kesesuaian Materi dengan Capaian Pembelajaran Lulusan pada Matakuliah Desain dan Strategi Pembelajaran,” *Prosiding Seminar Nasional IKIP Budi Utomo*, vol. 1, no. 01, pp. 524–528, Nov. 2020, doi: 10.33503/prosiding.v1i01.1141.

[2] O. Zlatkin-Troitschanskaia, H. A. Pant, and H. Coates, "Assessing student learning outcomes in higher education: challenges and international perspectives," *Assess Eval High Educ*, vol. 41, no. 5, pp. 655–661, Jul. 2016, doi: 10.1080/02602938.2016.1169501.

[3] H. Goss, "Student Learning Outcomes Assessment in Higher Education and in Academic Libraries: A Review of the Literature," *The Journal of Academic Librarianship*, vol. 48, no. 2, p. 102485, Mar. 2022, doi: 10.1016/j.acalib.2021.102485.

[4] D. Pembelajaran, D. Kemahasiswaan, D. Jenderal, P. Tinggi, D. Teknologi, and K. Pendidikan, "PANDUAN PENYUSUNAN KURIKULUM PENDIDIKAN TINGGI MENDUKUNG MERDEKA BELAJAR-KAMPUS MERDEKA MENUJU INDONESIA EMAS," 2024.

[5] N. S. Adilah, L. Hadjaratie, and R. Yusuf, "Pengembangan Sistem Informasi Rencana Pembelajaran Semester dan Evaluasi Capaian Pembelajaran Lulusan Berbasis Progressive Web App," *Diffusion: Journal of Systems and Information Technology*, vol. 2, no. 2, pp. 84–96, 2022.

[6] M. Rizky and T. Dirgahayu, "Implementasi Spreadsheet JExcel Pada Sistem Informasi Asesmen Capaian Pembelajaran Lulusan (Acapela)," *AUTOMATA*, vol. 1, no. 2, 2020.

[7] M. H. Zikry, A. B. Prasetijo, and R. Septiana, "Implementasi Front-End Sistem Penilaian Capaian Pembelajaran Lulusan (CPL) Dan Capaian Pembelajaran Matakuliah (CPMK)(Studi Kasus Pada Teknik Komputer)," *Jurnal Teknik Komputer*, vol. 3, no. 1, pp. 38–47, 2024.

[8] U. M. Ishaq, M. F. Wicaksono, and S. Nurhayati, "Aplikasi Probe Untuk Penilaian Capaian Pembelajaran Mahasiswa Pada Kurikulum OBE (Outcame-Based Education)," *Komputika : Jurnal Sistem Komputer*, vol. 12, no. 2, pp. 67–74, Sep. 2023, doi: 10.34010/komputika.v12i2.9763.

[9] V. P. Rantung, A. M. Dwisuprapto, and F. I. Sangkop, "Web-Based Application of Indonesia-Manado Translation Forum Using Extreme Programming Methodology," *CogITo Smart Journal*, vol. 10, no. 1, pp. 451–463, Jun. 2024, doi: 10.31154/cogito.v10i1.530.451-463.

[10] A. Andriani and J. F. Andry, "Designing a Web-Based Inventory Application at General Steel Supplier Using Extreme Programming Method," *CogITo Smart Journal*, vol. 9, no. 1, pp. 15–27, Jun. 2023, doi: 10.31154/cogito.v9i1.479.15-27.

[11] R. S. Pressman and B. R. Maxim, *SOFTWARE ENGINEERING: A PRACTITIONER'S APPROACH*, 9th ed. New York: McGraw-Hill Education, 2020.

[12] S. A. Asri, I. N. G. A. Astawa, I. G. A. M. Sunaya, K. A. Yasa, I. N. E. Indrayana, and W. Setiawan, "Implementation of Prototyping Method on Smart Village Application," *J Phys Conf Ser*, vol. 1569, no. 3, p. 032094, Jul. 2020, doi: 10.1088/1742-6596/1569/3/032094.

[13] E. Semenova, V. Tynchenko, S. Chashchina, V. Suetin, and A. Stashkevich, "Using UML to Describe the Development of Software Products Using an Object Approach," in *2022 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS)*, IEEE, Jun. 2022, pp. 1–4. doi: 10.1109/IEMTRONICS55184.2022.9795777.