

Object Oriented VS Structured Analysis and Design in System Development Courses

Aslina Saad, Nguarije Hambira, Harnani Mat Zin, Rasyidi Johan and Wang Shir Li

Abstract: Objectives: The main aim of this study was to investigate the factors that influence students', academicians', clients', as well as developer's preferences in choosing their preferred approach in system development, namely structured analysis design (SAD) or object-oriented analysis and design (OOAD). **Methods:** The research design was based on a survey methodology and a case study. For the survey, questionnaires were administered to 30 students and 38 academicians, who were randomly selected from several Malaysian universities. For the case study, the requirements of the information system were modeled and presented to several clients to elicit their feedback. The survey data were analyzed using SPSS **Findings:** The result shows that students preferred the use of OOAD approach, which clearly outnumbered those who preferred the SAD approach, which stood at 33%. Interestingly, the majority (53%) of academicians preferred the use of a mixture of both approaches. Likewise, the clients shared a similar view with the academicians, whereas the developer preferred the OOAD approach. **Application/Improvements:** Clearly, the findings suggest that both approaches are essential, but the one that is widely used by developers and preferred by students is OOAD, and thus should be given priority when it comes to structured analysis and design. As such, curriculum designers and institutions of higher learning, particularly those offering system analysis and design and related courses, should make the necessary changes to the existing curriculum such that the academic programs offered will be able to produce highly competent and skilled analysts and designers as required by the industry.

Index Terms: Analysis and Design, Object Oriented, Structured System development, Stakeholders' Preference

I. INTRODUCTION

Computer Science, Information Systems, and Software Engineering are among the popular academic programs offered at higher learning institutions worldwide. In these programs, System Analysis and Design, Information System Development, and Software Project are the compulsory courses that students have to master. Learning such courses

Revised Manuscript Received on September 22, 2019.

Aslina Saad, Computing Department, Universiti Pendidikan Sultan Idris, Tanjong Malim, Malaysia. aslina@fskik.upsi.edu.my

Nguarije Hambira, Computing Department, Universiti Pendidikan Sultan Idris, Tanjong Malim, Malaysia. nhambira@hotmail.com

Harnani Mat Zin, Computing Department, Universiti Pendidikan Sultan Idris, Tanjong Malim, Malaysia. harnani@fskik.upsi.edu.my

Rasyidi Johan, Computing Department, Universiti Pendidikan Sultan Idris, Tanjong Malim, Malaysia. rasyidi@fskik.upsi.edu.my

Wang Shir Li, Computing Department, Universiti Pendidikan Sultan Idris, Tanjong Malim, Malaysia. shirli_wang@fskik.upsi.edu.my

This work was supported by UPSI, "Geran Penyelidikan Khas Universiti Berteraskan Pendidikan" (2017-0294-107-01 KHAS PENDIDIKAN).

entail two approaches, namely structured and object-oriented analysis and design (A&D). Essentially, these approaches use two different sets of diagrams with different notations to model a system, which is presented to different stakeholders, such as users who will know the specific requirements for the system being modeled and programmers who will create the relevant codes for the system based on the system design. Given that there are two approaches available, academics are in dilemma in deciding which one is the more suitable method to be introduced at the university level. Currently, there are no formal guidelines that can help users select the better approach between the two.

In recent years, many researchers have investigated the application of object-oriented as well as structured analysis and design [1] - [4]. Although the object-oriented approach has been extensively used worldwide in this decade, empirical evidence to support the claim of its superiority over the structured approach in different phases of the system development is seriously lacking. In fact, different stakeholders may have different perspectives of the roles played by different diagrams in the system development process. Against this background, this study was carried out with the main aim of investigating the different stakeholders' preferences in using these two approaches in the different phases of system development. In this study, two types of stakeholders were recruited, namely students taking or who have taken system analysis and design-related courses and academicians who were in charge of teaching these courses. A key question that needed to be asked at this juncture was whether there was a need for a study that look at the approach of system analysis and design (SAD) through the examination of the difference in key stakeholders' preferences between OOAD and SAD.

Arguably, such a study is needed given that most of the studies that have been carried thus far only focused on either a certain group's perspective of SAD course content, time taken to complete a system based on either OOAD or SAD, or learning of programming by either approach [1], [5], [6]. Instead of studying these groups separately, a study combining these groups will be a better approach in examining the relationship between such groups, thus enabling researchers to answer the pressing question: "Which approach is the better to adopt in teaching such courses at the university level?". Surely, the findings of such a study can provide the essential guidelines for higher learning institutions in choosing the most appropriate approach to be

implemented in the learning of such courses at their respective institutions.

II. RESEARCH OBJECTIVES

The study was carried out with several objectives as follows:

- i. To investigate students' preferences in using the structured approach or object-oriented approach in modeling an information system.
- ii. To investigate academicians' preferences in using the structured approach or object-oriented approach in teaching and learning.
- iii. To model users' requirements using both the structured approach and object-oriented approach.
- iv. To investigate clients' preferences in using the structured approach or object-oriented to define the requirements of an information system.
- v. To investigate developers' understanding of the requirements of an information system to be developed.

III. METHODOLOGY

A quantitative approach involving an online survey was used in this study to collect the research data. The sample of the study consisted of 30 computer science and software engineering undergraduates and 38 academicians who randomly selected from several universities in Malaysia. Their feedback was elicited through a survey questionnaire (prepared in Google Form) consisting of several closed and open-ended questions. Specifically, they were asked to state their preferences in using the OOAD or SAD approaches, their justifications for such a use, and their knowledge of the OOAD and SAD concepts. Essentially, the questionnaire comprised three sections to collect information pertaining to their demographics, preferences in using either the OOAD approach or the SAD approach, and their knowledge. In addition, a qualitative approach via a case study involving selected clients and a developer was conducted. In this approach, two sets of diagrams consisting of comparable diagrams for both OOAD and SAD approaches were presented to the clients to elicit their specific preferences. As for the developer, the same two sets of diagrams were presented to elicit his specific preference. For the system development component, the researchers adopted the prototyping methodology because of its strong flexibility.

IV. FINDINGS AND DISCUSSIONS

As highlighted, this study was conducted to address four main objectives. Firstly, to identify factors that influence students' preferences in choosing a particular approach ((structured analysis and design (SAD) vs. object-oriented analysis and design (OOAD)). Secondly, to investigate factors that influence academicians' preferences in using either the SAD approach or OOAD approach. Thirdly, to investigate clients' specific preferences, and finally, to investigate a developer's specific preferences. The following are the details of the research findings.

A. Students' Preferences in Using the SAD Approach and the OOAD Approach

Five (5) questions were used to determine the students' specific preferences in selecting two types of diagrams, the answers of which showed their specific preference for either the SAD approach or the OOAD approach. Overall, the majority of the respondents indicated that they preferred diagrams that are under the category the SAD approach, as highlighted in their responses to three out of the four questions. In contrast, only a small minority of the respondents chose diagrams that are under the category of the OOAD approach. The following subsections discuss the detailed findings of students' specific preferences for diagrams of both approaches.

1) Students' preference: context diagram vs. use-case diagram

In answering several questions pertaining to their preference for either the context diagram or the use case diagram, the results indicated that the use case diagram was the most preferred between the two, as evidenced by 67% of the respondents who expressed such a preference. This finding is consistent with existing research findings, indicating an increase in the popularity of use case diagrams (under the OOAD approach) among practitioners, who have gradually been migrating from traditional approaches to this new approach [6], [7], [8].

A number of interesting themes emerged from the respondents' responses when asked for their justifications for choosing the use case diagram instead of the context diagram. Out of 20 respondents (who preferred the use case diagram), 12 justified their preference by stating that the approach was "easy to understand". Unfortunately, a large majority of these respondents did not provide any details as to why they made such a preference. However, a few of the respondents did provide some reasons to explain such a choice, as highlighted in their responses, such as "detailed description of the system", "easy to see the flow of the system from the perspective of users", and "easy to understand as it visually represents stakeholder and process relationship", which collectively emphasizes the attributes of such a type of diagrams to facilitate comprehension.

In addition, another reason cited for the preference of the use case diagram over the context diagram was attributed to the former's simplicity and its use of real-life objects, which was supported by the respondents' responses, such as "simplicity and the fact that it is related to real life objects". Other reasons given by the respondents included its flexibility (by inferring their response with regard to the "include and exclude" aspect of such a diagram) and its high level of abstraction. Interestingly, these findings are consistent with the finding of a study carried out [2], who found that OOAD was a realistic approach in designing and implementing systems with improved security by means of encapsulation, abstraction, and polymorphism.

2) *Students' preference: data flow diagram (DFD) vs. sequence diagram*

In responding to several questions pertaining to their preferred diagram for either the data flow diagram (DFD) or the sequence diagram, the results showed the former was the most preferred diagram, with 63% of the respondents who expressed such a preference. Such a finding was not surprising, given that, lately, most software design and development practitioners have been employing the OOAD approach. Nevertheless, a sizeable number of companies are still relying on the SAD approach for the system design and development [9]. Furthermore, the reason why they did not prefer the sequence diagram was that it has to be constructed for each use case of a given system, making the process laboriously tedious and complicated.

For instance, if a given system has 5 use cases (create new user, check status, request transfer, chat with admin, and upload documents), an analysts using this approach will be required to draw 5 sets of sequence diagrams with all relevant classes as well as to-and-fro message passage to show each of these use cases in more details. This detailed process, seemingly easy in the implementation phase, can be quite cumbersome to execute, which probably prompted the students to share the same feelings. Out of 19 respondents (who preferred the use of DFD), nine (9) justified their preference by stating that the diagram was “easy to use or easy to understand”.

Likewise, a large majority of these respondents did not provide any details as to why they made such a preference. Nonetheless, some of the respondents' responses, such as “easy to understand” (having the highest frequency), “its ability to show the system logic of complex systems”, “more details on data manipulation” and “ensure which data are the most needed in the process”, may partly explain their preference of the OOAD approach.

3) *Student preference: flowchart vs. activity diagram*

Between the flowchart diagram and activity diagram, 60% of the respondents indicated that they preferred the former more than the latter. Their justifications for preferring the flowchart were primarily attributed to its ease of use, with 11 out of 18 respondents expressing a revealing response, namely “ease of use/easy to understand”. Similar to the earlier findings, not much elaboration was offered as to why the respondents felt it was easier to understand the flowchart more than the activity diagrams. However, a closer scrutiny of a few of their responses may provide some clues, such as “familiar”, “flowchart allows the visual representation of the sequence of steps and decisions needed to perform a system process”, and “allows for the developer to adjust with the use of a draft prior to system development”.

4) *Student preference: entity relationship diagram (ERD) vs. class diagram*

Between the entity relationship diagrams (ERD) and the class diagram, the results indicated that the use of the former was the most preferred, with 63% of the respondents indicating such a preference. Again, such a finding is consistent with existing research findings, which emphasize the importance of ERD in designing a sound database that allows for easy storage and retrieval of data [9], [10]. A few

interesting themes emerged from their responses when they were asked for the justifications for such a preference. Nine (9) out of 19 nineteen respondents made a few responses that may shed some light on the reasons that might have prompted them to prefer ERD, such as “easy to understand”, “easy for collaboration and the foreign key”, and “ERD is simple compared to class diagram”. Likewise, their responses, such as “ease of data classification”, “the fact that it is database-oriented”, and “ERD is simplified from database”, may further explain their tendency to prefer ERD. In addition, several respondents noted that ERD could be converted it into a database when working with MySQL Workbench. Quite revealingly, most respondents indicated that their confidence was high when using ERD compared to that of the class diagram.

5) *Students' overall preference: structured analysis and design (SAD) vs. object-oriented analysis and design (OOAD)*

Interestingly, the findings of this study were rather unanticipated as an overwhelming majority of respondents (67%) indicated that they preferred the use of the OOAD approach more than the SAD approach. Such findings are consistent with those of recent research [6], [7], [9], [11], [12], indicating an increase in the popularity of the OOAD approach for major system analysis and design projects. However, these findings were unanticipated as the majority of the respondents preferred the use of diagrams based on the SAD approach more than those based on the OOAD approach. As such, it would be logical to expect they would have chosen the SAD approach as the preferred approach, which was not the case as demonstrated in this study. Hence, such conflicting findings (respondents' preference for diagrams based on the SAD approach against their preference for the OOAD approach) entail some explanation.

Out of 20 respondents (who preferred the OOAD approach), seven (7) cited “ease of understanding” as their primary reason in choosing this approach. In addition, they indicated that from their experience “clients seemed to understand the illustrations of a system far better with the use of use case” compared to other design illustrations. Overall, with the exception of this finding, the remaining findings of this study are consistent with previous findings reported in the current literature. More tellingly, the respondents' preference for such an approach can be best explained by their responses as follows: “objects are reusable, flexible approach, reuse of code, and security enhancement, such as polymorphisms, abstraction, and inheritance”. Furthermore, the use of real-world objects could also be another reason that influenced such a preference. Collectively, most of the reasons indicated by the respondents are similar with those reported in the existing literature [7], [8], [10], [11], [13] - [17].

B. Students' level of understanding

To examine students' understanding of the OOAD and SAD concepts, a questionnaire consisting of 12 true-false questions was employed. Specifically, the aim was to elicit information on the extent to which the respondents had

understood the concepts related to system analysis and design taught in the courses. With such information, the students' understanding of the concepts learned in the courses could be compared with their reasons or justifications for their specific preference for the approaches. Arguably, strong content knowledge of the courses would correlate significantly with their justifications in choosing a particular approach.

Overall analysis of their responses revealed that there was a fair mixture of responses with regard to their views of both approaches. For the statement that data flow diagram (DFD) helps illustrate the movement of data between external entities and processes as well as data stores within a system, 96.7% of the respondents indicated that such a statement was true. In contrast, the remaining 3.3% indicated such a statement was false. Likewise, the majority of respondents (90%) indicated that it was true that context diagrams depict system boundaries that interact with a system; whereas only three respondents (10%) indicated otherwise. Similarly, for the use case diagram, a large majority of respondents (85.7%) agreed with the given definition of use case diagrams, as opposed to only 13.3% who indicated such a definition was false.

As for the question that asked if use case could be further explained by a sequence diagram or not, the majority of respondents (87%) indicated that such a statement was true; whereas only a small percentage of respondents (13%) indicated otherwise. Likewise, most of the respondents (97%) chose the 'true' response for the question that asks whether or not the activity diagram depicts the conditional logic of the steps of system activities in completing a business process. A similar trend was observed with the question asking whether or not the sequence diagram highlights the interactions among objects over a certain duration, with most of the respondents (87%) agreed that such a statement was true; whereas only 13% indicated that such a statement was false. More revealingly, the question that asked whether the context diagram is comparable with the activity diagram, 60% of the respondents answered it incorrectly.

For the question with the statement that the collaboration diagram and the sequence diagram were identical, the majority of responses (83%) indicated that such a statement was true as opposed to a small minority (17%) who indicated otherwise. Likewise, the majority of respondents (80%) noted that the statement claiming that the class diagram represents the same thing as that of the entity relationship diagram (ERD) was true, with only a small minority (20%) claiming it to be false. Similarly, the same trend was observed with the question asking whether DFD 1 is the detailed process of DFD 0 or not, as evidenced by a majority of respondents who noted that such a statement was true. Finally, for the last question with the statement about the functional primitive, slightly more than half of the respondents (67%) indicated such a statement to be true. Generally, the respondents managed to answer with a high degree of accuracy, with 80% of their answers being accurate, indicating that they had a firm grasp of the important concepts of the subject matter.

C. Academicians' preference

In this study, 38 academicians were selected for the survey, of whom 10 (26.3%) were males and 28 (73.7%) were females. In terms of nationality, the majority of the

respondents (92.1%) were Malaysians and the remaining 7.9% were Namibians. In terms of age, more than half of the respondents ($n = 24$ or 63.2%) were in the age range of 35 – 44 years, while 9 (23.7%) were in the age range of 25 – 34. In contrast, those in the age range of 45 – 54 and less than 24 years were represented by 3 respondents (7.7%) and 2 respondents (5.3%), respectively. Almost a third of the respondents (31.6%) had teaching experiences in the range of 11-15 years, closely followed by 28.9% of the respondents who indicated they had teaching experiences in the range of 11-15 years. Those with teaching experiences less than 5 years were represented by 18.4% of the respondents. For the use of approaches, more than half of the respondents (53%) reported that they preferred the use of a mixture of both approaches (structured and object-oriented), and almost a third of the respondents (31%) said they preferred the object-oriented approach. In contrast, only 16% indicated they preferred the structured approach.

As to which approach was most preferred by clients, 58% of the respondents said it was the object-oriented approach. Furthermore, the majority of the respondents (76%) indicated that such an approach was the approach that programmers could easily understand. Similarly, with regard to which method that could be easily understood by various stakeholders, slightly more than half of the respondents (55%) indicated that such a method would be the object-oriented approach. Almost, two-thirds of the respondents (74%) indicated that they believed that analysts would most likely prefer the object-oriented approach. Likewise, the majority of the respondents (82%) indicated that software houses would prefer the object-oriented approach.

Additionally, the academicians were given a chance to provide their input. For instance, one of the respondents said that "there is a tool that can automatically convert OOAD to OOP". Apparently, this unique tool may partly explain why most developers prefer to use OOAD. Another respondent responded by asserting that "users prefer the structured method to help them better understand the workflows, steps, and requirements, and developers also prefer such a method for reuse purposes." This response further reinforced the preference of practitioners to adopt such an approach, evidently made clear by the following assertion: "the paradigm really depends on the problem-solving basis, especially for students who are new to programming to learn the procedural elements that can help them develop the coding skills; then, an injection of new paradigm is absolutely needed so individuals can see the differences between the two approaches such that they can decide which one is more practical". Furthermore, another respondent's response was tellingly revealing when he said: "in terms of usability, the structured approach works best, but for a better understanding of a more complex system, the object-oriented approach is a must." Nevertheless, one of the academicians stated that "students like to use the structure-based approach because they are weak in programming using the object-approach".

As revealed, the majority of the academicians (53%) preferred the use of both the structured approach and object-oriented approach. Such a specific preference can be explained by some of their responses, such as “the diagrams are specific for certain purposes”, “the approach is easy to understand”, “it will take less time to apply it”, “it is cheaper to implement this approach”, and “it is easily understood by students”. Nonetheless, it needs to be emphasized that this finding does not clearly explain why the combination of both approaches is cheaper or easier to be implemented. With the exception of this finding, the rest of the findings are consistent with previous findings, highlighting the popularity of the object-oriented approach. Based on the academicians’ responses, it is clear that clients would prefer the use of the object-oriented approach because they could better understand the validation of the system requests. Likewise, the same holds true for programmers and analysts of software house and systems who would have the same preference.

D. Clients’ preference

The clients’ requirements of the system were modeled using both approaches. The two approaches were compared in relation to the respective purpose of the diagrams. For example, the use case diagram (OOAD) and the context diagram (SAD) were used to define the scope of a system. In addition, the pair of data flow diagram (SAD) and sequence diagram (OOAD) was used to define the details of a system’s function. To compare users’ preferences in defining the flow of a system, activity diagram (OOAD) and flow chart (SAD) were used. Fig 1-3 represent the models of the three sets of diagrams.

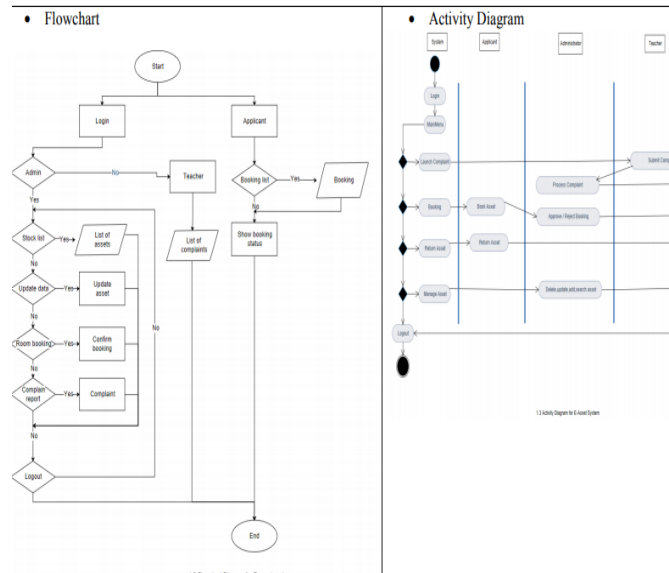


Fig. 3. The flow chart and activity diagram of a system

The finding of the interview with the clients revealed that they preferred the OOAD approach to define the scope of a system with the application of the use case diagram. However, to explain the details of each function of the system, they would prefer to use DFD, which is a diagram used in the SAD approach, rather than a sequence diagram. In fact, they would be able to discern better the flow of the system functions with the use of the activity diagram (OOAD) compared to the use of the flow chart. Hence, from the clients’ perspective, they tend to prefer the mixed approach, the preference of which is similar to that of the academicians.

The developer was also given the same set of diagrams that typically serves as a specification document in developing a system. Specifically, he was asked about his preference in using either object-oriented diagrams or structured-based diagram. Overall, he preferred the use of diagrams that are based on the OOAD approach. The key factor that motivated his preference for this approach was that with the OOAD approach diagrams, the transition from the design phase to the implementation (coding) phase could be carried out quite effortlessly, given that the OOAD concepts allow for easy integration with the corresponding programming language. Admittedly, the use of sequence diagrams, instead of its equivalence (data flow diagrams) in the structured analysis and design (SAD) approach, would result in many diagrams to represent each use-case diagram, which entails more time is needed to complete a project. Regardless of this constraint, he still preferred the use of the OOAD approach, as the drawing of many diagrams for each use-case, in his opinion, provides better documentation as a whole.

V. CONCLUSION

The findings of this study carry profound implications on the current practice, affecting curriculum designers, educators, students, institutions of higher learning as well as system analysis and design researchers in ways identified as follows:

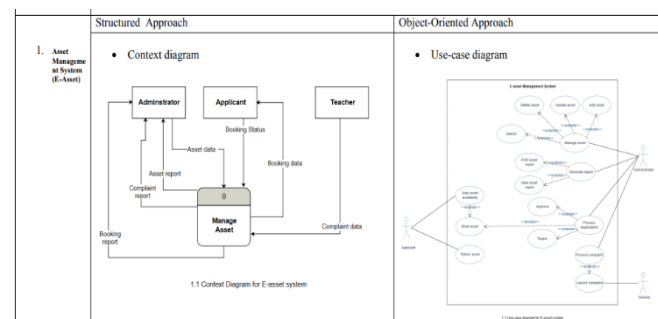


Fig. 1. The context diagram and use diagram of a system

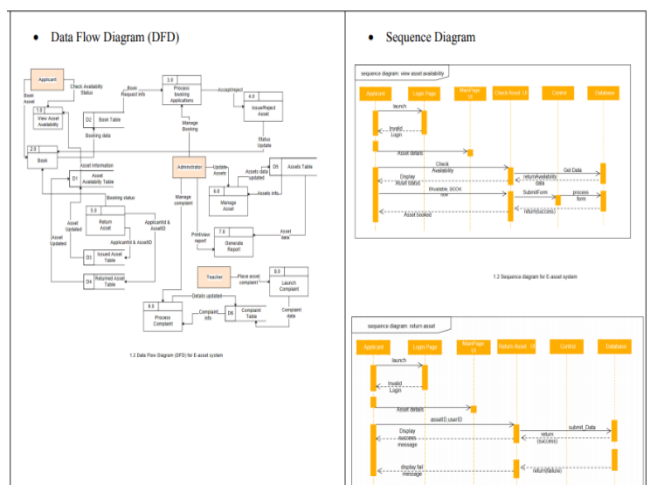


Fig. 2. The DFD and sequence diagram of a system

- Curriculum designers need to take into consideration the opinions as well as inputs of the most important stakeholders of them all – the students. As revealed, a number of factors would influence their preference for a particular approach; thus, it is of paramount importance that curriculum designers focus on such factors and implement the findings in the curriculum.
- Academicians should take into account such findings in helping them to choose the most preferred approach that will best serve the needs and interest of their students.
- Institutions of higher learning offering system analysis and design and related courses should make the necessary changes to the current curriculum to ensure future analysts and designers will have sufficient knowledge and skills as required by the industry.
- Researchers in this discipline should further investigate the intricate relationship between a chosen approach and its corresponding diagrams.

As demonstrated in this study, students preferred to use the object-oriented approach over the structured approach, whereas academicians were inclined to use the combination of both approaches. As such, common ground has to be found to accommodate such differences such that both parties will be able to learn and teach with greater efficacy, respectively. Likewise, clients too preferred a model that incorporated a mixture of both approaches. In contrast, developers preferred the OOAD approach. Arguably, students will have to rely on the structured analysis and design approach to understand the basics of system analysis and design, the impact of which can help them learn how to develop systems that meet clients' requirements. Therefore, opportunities should be given to students to choose an approach they best understand, which they can easily integrate into their future projects, saving precious times of all stakeholders involved. In summary, both approaches are essential, but the one that is widely used by developers and preferred by students is OOAD, and should be given priority when it comes to teaching structured analysis and design courses.

REFERENCES

1. Benjamin, A.D. M (2014, 1 August 2017). Object Oriented Vs Structured Analysis: Is Structured Analysis Dead? Retrieved from <https://www.linkedin.com/pulse/20141023035917-338627392-object-oriented-vs-structured-analysis-is-structured-analysis-dead>
2. Falessi, D., Cantone, G., & Grande, C. (2007). A Comparison of Structured Analysis and Object Oriented Analysis an Experimental Study.
3. Motaz, S. (2010, 28 July 2017). Structured vs object oriented analysis and design. Retrieved from <https://www.slideshare.net/mksaad/structure-vs-object-oriented-analysis-and-design>
4. Pefkaros, K. (2008). Using object-oriented analysis and design over traditional structured analysis and design. *International Journal of Business Research*, 8(2), 219-227.
5. Guidry, B. N., & Stevens, D. P. (2014). Comparing perceptions of the systems analysis and design course. *Journal of Computer Information Systems*, 55(1), 40-47.

6. Korinek, O., & Hubalovsky, S. (2017, September). Research of Methods of Learning of Programming Objects-First and Object-Later. In *International Conference on Applied Physics, System Science and Computers* (pp. 183-189). Springer, Cham.
7. Acheson, P. (2010, April). Methodology for object-oriented system architecture development. In *Systems Conference, 2010 4th Annual IEEE* (pp. 643-646). IEEE.
8. Kuo, T. C., Hsu, C. W., Ku, K. C., Chen, P. S., & Lin, C. H. (2012). A collaborative model for controlling the green supply network in the motorcycle industry. *Advanced Engineering Informatics*, 26(4), 941-950.
9. Sharma, N., & Sawai, D. (2011). Suggested Methodologies for Enhancing SAD. In *2011 Fourth International Conference on Emerging Trends in Engineering & Technology* (pp. 1-4). IEEE.
10. Ang'ondi, E. K. (2013, July). Teachers Attitudes and perceptions on the use of ICT in teaching and learning as observed by ICT champions. In *X World Conference on Computers in Education* (pp. 1-8).
11. Ghoreshi, M., & Haghghi, H. (2016). An incremental method for extracting tests from object-oriented specification. *Information and Software Technology*, 78, 1-26.
12. Sinha, A. P., & Jain, H. (2017). Reusing business components and objects for modeling business systems: The influence of decomposition characteristics and analyst experience. *Journal of Systems and Software*, 131, 550-569.
13. Abdoli, S., & Kara, S. (2016). Designing warehouse logical architecture by applying object oriented model based system engineering. *Procedia CIRP*, 50, 713-718.
14. Bonvini, M., & Leva, A. (2011, January). Object-oriented quasi-3D sub-zonal airflow models for energy-related building simulation. In *18th IFAC World Congress (IFAC'11)* (pp. 12946-12951).
15. Evertsz, R., Thangarajah, J., Yadav, N., & Ly, T. (2015). A framework for modelling tactical decision-making in autonomous systems. *Journal of Systems and Software*, 110, 222-238.
16. Fletcher, K. K., & Liu, X. (2011, June). Security requirements analysis, specification, prioritization and policy development in cyber-physical systems. In *Secure Software Integration & Reliability Improvement Companion (SSIRI-C), 2011 5th International Conference on* (pp. 106-113). IEEE.
17. Sarkar, S., & Negi, A. (2013). Designing a high quality online course: A process driven approach using UML. In *Contemporary Computing (IC3), 2013 Sixth International Conference on* (pp. 262-267). IEEE.

AUTHORS PROFILE



Dr Aslina Saad is working as a Senior Lecturer in Faculty of Art, Computing and Creative Industry. She is a graduate in Bachelor of Information Technology (Science and System Management) and Master of Science (Information Technology). She received her PhD in Computer Science (Information Systems) from University of Loughborough and is currently attached to University Pendidikan Sultan Idris. Her research interests are on Computer in Education, Management Information System, Artificial Intelligence and Knowledge Management. Currently, she is teaching System Analysis and Design, Software Design, Software Project and Information System Development courses.



Nguarije Hambira is an aspiring young researcher with an immense interest in the area of information technology and its integration in the education sector. He obtained his honors degree from Sultan Idris Education University in 2017 and has gone on to publish over 5 conference proceedings and journal papers. He is presently pursuing his master's degree in education with specialization in information technology. He has extensive experience in research and publication gained while working as a research assistant during his undergraduate as well as postgraduate studies at Sultan Idris Education University.



Harnani Mat Zin is working as a Lecturer in Faculty of Art, Computing and Creative Industry. She is a graduate in Bachelor of Computer Science and obtained her Master of Science (Computer Science) degree from Universiti Sains Malaysia. Her research interests are on Information System (IS), Information Retrieval, Open Source Software, Data Mining, Opinion Mining.



Rasyidi Johan is working as a Lecturer in the Faculty of Art, Computing and Creative Industry. He holds a Bachelor of Computer Science and obtained his Master of Science (IT) degree from UTM. His research interests are on Multimedia Courseware Design and Development, Web Based Personalize Learning Application, Framework

Design for Learning Environment, Artificial Intelligence in Education, Open Learner Modelling as well as Spatial Abilities.



Wang Shir Li was born on July 14, 1979 in Selangor, Malaysia. She received the B. Tech. (Hons.) in quality control & instrumentation, in 2002 and M.Sc. in electrical & electronic engineering, in 2007, both from the Universiti Sains Malaysia, Penang, Malaysia. She obtained her Ph.D. degree from the University of New

South Wales, NSW, Australia, in 2012. She is currently a senior lecturer at the Faculty of Art, Computing and Creative Industry, Universiti Pendidikan Sultan Idris. Her research interests include machine learning, data mining, optimization, and adversarial learning. She is a Member of the IEEE since 2013.