IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 7 December 2022, accepted 18 December 2022, date of publication 28 December 2022, date of current version 5 January 2023.

Digital Object Identifier 10.1109/ACCESS.2022.3232947

RESEARCH ARTICLE

Checklist Approach for the Development of Educational Applications by Novice Software Developers

NORZILA NGADIMAN[1], (Member, IEEE), SHAHIDA SULAIMAN[1], (Member, IEEE), NORSHAM IDRIS[1], (Member, IEEE), MOHD RAZAK SAMINGAN[1], AND HASNAH MOHAMED[2], (Member, IEEE)

[1]Faculty of Computing, Universiti Teknologi Malaysia, Johor Bahru, Johor 81310, Malaysia
[2]Faculty of Social Science and Humanities, School of Education, Universiti Teknologi Malaysia, Johor Bahru, Johor 81310, Malaysia

Corresponding author: Norzila Ngadiman (norzila.ngadiman@ieee.org)

The work of Norzila Ngadiman was supported by the South East Johor Development Authority (KEJORA) through the Scholarship funded by the Centre for Advancement in Rural Education Informatics (iCARE) Universiti Teknologi Malaysia (UTM)-KEJORA through Cost Centres under Grant 4B349 and Grant 4C555.

ABSTRACT Innovation and challenges are significant factors that lead to the improvement in technology involving various sectors, including the educational field. New methods and techniques have been introduced in teaching and learning among learners and educators. Modern technology generates an effective learning process that increases the students' interest and understanding of learning activities. Hence, software engineers need to develop high-quality educational applications that include the required elements such as learning materials and types of assessments. In addition, it should align with stipulated guidelines, timelines, budgets, and policies. Adaptation of a checklist approach in developing educational software can improve the quality of the software significantly. The checklist approach appears to be more effective for developers, especially novices with limited knowledge in relevant evidence-based principles. This study investigates the checklist approach for novice software developers in developing educational applications. A survey was employed to gather data from a group of respondents. This study used two types of sampling methods which were purposive sampling and Yamane formula. It shows that 89.19% of the respondents had understood and benefited from the provided checklist.

INDEX TERMS Novice software developer, educational software application, checklist, software quality, functional requirement, non-functional requirement.

I. INTRODUCTION

One of the main issues in education study is the challenges in finding a better solution to make the education revolution more effective [1]. Revolution in education is non-linear, complex, and dynamic as the world progresses due to technological innovation and improved interconnectedness among cultures and nations [2]. Innovation and challenges involve many aspects, including learning techniques and human resources. Information and communication technology (ICT) provides new methods of teaching and learning [1]. Hence, the revolution in ICT has made it possible for teachers and students to collaborate and communicate in diverse ways [3].

Implementation of educational software applications such as teaching tools generates potential benefits for learners and educators to share and easily access learning materials [3]. Software developers need to develop high-quality educational applications that meet both functional requirements (FRs) and non-functional requirements (NFRs) for diverse level of learners. For instance, developers should consider how students can retrieve learning materials and perform the assessments based on their levels and/or ages, besides they can easily understand how to use the applications.

The associate editor coordinating the review of this manuscript and approving it for publication was Liang-Bi Chen.

The educational applications should align with the stipulated guidelines, timelines, budgets, and policies [4]. Some studies prove that checklists can solve concerned problems [5], [6] and the quality of software improves significantly based on the checklist approach [6] that can be applied in developing educational software. In addition, the development of effective educational software is based on rules or scenarios provided by users or stakeholders while new software developers should know how to refer to and use the items listed in a checklist [7].

When software developers especially beginners with less experience, skills, and knowledge are assigned to develop an educational application with minimal supervision, it is crucial for them to understand that the application must comprise several educational features mainly contents and materials, including their formative and summative assessments. Besides, novice software developers (NSDs) need to focus on software quality to increase user satisfaction and loyalty, and application requirements must be verifiable as individual features, either as FRs or NFRs, at the system level [8].

If software developers, mainly NSDs overlook these significant aspects in software development, the deployed applications may not meet expected target. Therefore, this research aims to investigate the issues faced by software developers, especially among NSDs. Recent works in Table 1 list the research concerns and respective challenges in software development with less experience like NSDs.

TABLE 1. Challenges in software development.

Research Concern	Challenges
Serverless Computing-Based Applications [9]	Developers tended to focus on the application logic, which resulted in incorrect system variables defined. Furthermore, they only focused on a particular process without considering other constraints.
Software Security Engineering [10]	Developers had confidence in their ability to complete secure-development tasks, such as identifying security problems during software design or using secure programming languages. However, the developers of these platforms were less experienced, so reaching the sample used in this study required significant effort.
Medical Software Devices [11]	Identifying and comprehending the regulatory requirements applied to the software development process proved difficult for software developers. Furthermore, unclear regulatory requirements played a significant role in the software's regulatory compliance, especially when the requirements involved legal text.
Edge Computing [12]	Software developers (such as, applications/content providers, innovators, and startups) of an edge hosting environment had to gain a better understanding of many aspects of all deployment options offered by the edge, from benefits and constraints, to gains and cost tradeoffs.
Framework for Sustainability of Software System Design (FSSSD) [13]	It was difficult to contemplate techniques to connect the first, second, and third order impacts in each sustainability aspect, because it was highly dependent on each individual's perspective.

Generally, software developers, especially NSDs, do not realize the importance of NFRs during application development, which consists of multiple functions. They should be aware and define the NFRs at an early stage to avoid some latent problems, which involve quality characteristics in application developments, including educational applications [14]. NFRs refer to the quality of an application that can be defined from various aspects; for example, educational applications must include the presentation of learning materials, assessment functions, and tracking of students' achievements in a way that motivates users and increases their level of satisfaction and loyalty. The quality in educational applications allows them to become more popular nowadays, but some of them do not offer appropriate user interaction, which will also lead to a bad user experience due to the difficulty to use, learn and understand the applications [15].

Software developers, especially NSDs, need guidance to refer to during the development stage. Although they may be proficient in adapting FRs to the latest technology and deciding the basic features for the application, they sometimes fail to recognize the application quality that needs to be applied [16]. NFR is a tacit knowledge that users find difficult to express, particularly during the elicitation process. Therefore, a user's description sometimes seems ambiguous and unclear, resulting in the software developer to interpret requests in a variety of ways, due to the unfamiliarity of the NFR aspects [16].

Besides considering the quality of software applications, a method to present them should be defined by considering the latest technology in the current environment, such as the implementation of attractive learning methods, which is more effective compared to traditional methods [17]. Quality educational applications have been inspired and satisfied in some genuine administered learning assignments [18]. Furthermore, students can transform into becoming progressively inquisitive and rationally inspired. This will increase their appropriateness recognizability and understandability during the learning process [19].

Therefore, the aforementioned aspects are generally faced by NSDs. They may understand how to adapt the latest technology, but they may not be clear on how to adapt the technology with an effective quality application mainly for those who are still new in developing educational software. Hence, this research aims to provide a checklist-based guideline for NSDs to refer to, in order to gain better knowledge of FRs and NFRs, which may assist in the development of educational applications.

A checklist is defined as a "list of action items, tasks, or behaviors arranged in a consistent manner, which allows the evaluator to record the presence or absence of the individual listed item" (p. 24) [8]. Generally, a checklist uses extensive taxonomically-related or cognitive aids that diverge in execution, function, and use [20]. This study applies the 'Breakthrough Collaborative' concept in the checklist tool [7]. The collaborative aspect combines all necessary items, tasks, or rules in the learning activities. Besides, the breakthrough element focuses on collective trust and understanding in a community with a culture centered on empowerment, connection, and fun. The novel collaborative concept used in this study consists of the application components and sub-components to be included in the checklist.

The checklist is used to obtain knowledge on what is to be included in the learning cycle. Thus, in the development of an educational application, software developers, especially NSDs can benefit from the checklist, whereby a checklist is a useful tool for non-expert professionals that can help to improve the methodological quality of the deliverables [21]. Furthermore, a checklist helps produce results that are more detailed, comparable, and replicable to the educational intervention descriptions. A checklist can also be used for reporting, as well as a support tool to guide people [22].

Thus, this study examines the checklist approach for NSDs to develop educational software applications by implementing the predefined quality characteristics. The following Section 2 examines the background of the checklist approach, including recent studies and the implementation of the International Organization for Standardization (ISO). Section 3 elaborates the use of checklist in specific software development methodology, its development, and the details of the proposed checklist approach. Section 4 explains the implementation of the checklist by the NSDs while Section 5 reports the evaluation. Finally, Section 6 provides key takeaways from this study and possible future works.

II. BACKGROUND OF CHECKLIST APPROACH

A checklist involves a process of listing a task or a process that can be used as a guidance. The listed items can be grouped by comparing specific factors by category, format, or characteristics. Hence, the checklist simplifies tasks as tasks can be managed and reduced to simpler sub-tasks [7] and increase the 'reliability' of various assessment categories [6].

Guidelines in the form of a checklist will be more effective when checking tasks or items one by one with explicit monitoring [7]. This section comprises two subsections that are the implementation of the checklist in ISO documents and how other researchers have adopted the checklist approach in their works.

A. ISO DOCUMENTS

ISO is a standard document which individuals from various fields and sectors use as a reference to ensure the success of their tasks and projects from start to finish. This study reviewed eight ISO documents that used the checklist approach as listed in Table 2.

TABLE 2. Implementation of the checklist approach in iso documents.

ISO Document	Description
ISO 14001:2015 [23]	The Gap Analysis Tool Checklist provides two types of self-assessment checklists: i. An overview of current compliance, such as an online comparison tool. ii. Details the exact standard requirements needed by going over every requirement.
ISO 9001:2015 [24]	The Internal Audit Checklist confirms that the specified requirements have been followed and adequate evidence has been recorded.
ISO 9004:2018 [25]	The Self-Audit Checklist identifies areas that need to be improved and prioritized based on business goals or regulatory standards.
ISO 45001:2018 [26]	The Internal Audit Checklist helps in auditing requirements systematically as a beneficial reference point for before, during, and after the audit process.
ISO 22000:2018 [27]	The Audit Checklist is a tool used to prepare implementation and assessment of the Food Safety Management Systems (FSMS). It is suitable for use in the last stage of the internal preparation for the external auditor certification.
ISO 17025:2017 [28]	A checklist tool to determine if a laboratory meets the required competencies for testing and calibration set by the ISO 17025:2017 standard.
ISO 27001:2014 [29]	A checklist tool to determine if an organization meets the requirements of the international standard for implementing an effective Information Security Management System (ISMS).
ISO 13485:2016 [30]	An audit checklist for quality managers to determine whether the QMS of an organization aligns with the ISO 13485:2016 standard.

Therefore, checklists in ISO documents of professional bodies serve as tools to ensure the systematic execution of all tasks and specified requirements are satisfied [31].

B. PREVIOUS RESEARCH ON CHECKLIST APPROACH

A checklist is useful to determine the predefined tasks "for which the particulars of context are immaterial, reliance on human memory is a known problem, and variations in those procedures are undesirable" [8]. Thus, checklists serve as a reference even when the users are aware of proper procedures and assists stakeholders to assess a paper against the "criteria for compliance with the scientific method." [7].

Normally, a checklist is an audit tool to highlight discrepancies in 'work-as-done' that is to check whether things have been done in an ideal scenario [8]. Higgins and Boorman [32] introduce different types of checklists and their application in diverse fields as listed in Table 3.

TABLE 3. Types of checklists [30].

Type	Definition	Example
Procedural checklist	Critical tasks (lengthy or complex), performed irregularly to ensure certain tasks are successfully performed.	Assembly instructions for furniture.
Preparation checklist	Ensures execution of all steps in multistep procedure and in order of execution.	Shopping list
Problem-solving checklist	Sequential, multistep procedure with decision points used to support documentation.	Medical protocol
Prevention checklist	Enlist cause of errors, high-risk, or high-hazard conditions that may result in injuries or death of users.	Equipment setup checklist
Forms of execution		
Do-confirm	The checklist confirms that no steps or items were missed when tasks are performed from memory.	
Read-do	The checklist enables the performance of infrequent tasks not committed to memory.	

A checklist ensures the performance of tasks in the order specified, the last being the least important [32]. A conceptualizing checklist performs or identifies a distinction referred to as "do-confirm" (do it, confirm it) versus "read-do" (read it, do it) [9]. Clinical staff use a checklist as a communication tool for 'check-and-challenge' [6].

In the medical field, formalizing tasks and work processes in the form of a checklist places increased emphasis on 'work-as-imagined' (how things would be done in an ideal scenario), which some hospital teams found difficult to reconcile with 'work-as done' in the messiness of everyday practice [8]. Clinical areas also use a safety checklist to reduce medical errors and improve patient safety.

Individualized checklists may be produced based on information from the patient's medical records while also considering the context of the clinical workflow [5]. The checklist is a 'technical' tool that helps to improve communication between junior and senior staff across specialties and encourages them to complete assessments to avoid any clinical difficulty in the complex sociocultural ways of working [6].

Economic sectors such as engineering, medicine, and aeronautics use checklists to ensure compliance with guidelines. In court cases, non-compliance with operational checklists or evidence-based guidelines may lead to apportionment of blame for dismissal or payment of damages [7], and limitations in assessments [6]. In addition, the successful implementation of checklists also depends on good teamwork and communication as the prerequisites [20].

The successful implementation of checklists depends on the defined role of hierarchies and professional boundaries s [23]. In the 1930s, the checklist approach proved great potential in the aviation industry. Since then, it has been increasingly used in the medical industry as a cognitive aid to improve clinical processes and outcomes [8].

III. CHECKLIST APPROACH IN SOFTWARE DEVELOPMENT

There are five phases in the software development life cycle (SDLC): (1) requirement analysis, (2) design, (3) implementation, (4) testing, and (5) deployment and maintenance. Thus, Agile techniques in the SDLC promote adaptive development by indicating the sprint cycles of planning, action, rectification, and change to improve results [33].

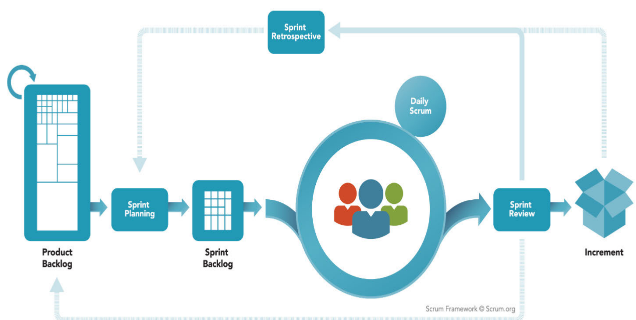

FIGURE 1. Agile scrum methodology [34].

In this study, the applied Agile Scrum Methodology is one of the most common applications associated with agile principles, as illustrated in Fig. 1 [34]. Agile Sprint is a powerful and helpful tool for developers to define the structure of processes and enable a wide range of beneficial and creative explorations [35].

In Agile methodology, NSDs need to divide their tasks based on each sprint. In our study, there are four sprints to complete, and each sprint takes two weeks to complete. As shown in Figure 1, NSDs need to present their progress to the product owner every two weeks to comply with the project's goals. Hence, this section elaborates the steps in creating the checklist and presents the proposed checklist.

A. DEVELOPMENT OF CHECKLIST

Table 4 presents eight steps in developing a checklist for this study. The theoretical framework (see Fig. 1) constitutes the baseline for the software application development checklist for NSDs.

TABLE 4. Steps for producing a checklist.

No.	Step	Description
1	Research scope	To propose a checklist as a guide for NSDs in developing an educational application.
2	Research audience	Software developers include novices and senior developers. The users of the software are teachers and students.
3	Theoretical framework	A diagram that represents an initial idea on the research overview (see Figure 2).
4	Quality characteristics	Based on the theoretical framework, a set of questionnaires were produced, distributed, and analyzed. The Goal Quality Measure (GQM) [36] approach defines the metrics to be validated (see Figure 3). From the analysis and findings, quality characteristics such as user interface aesthetics, appropriateness recognizability, learnability, satisfaction and effectiveness were selected to be included in an educational application.
5	Comparative	Comparisons of four recent online educational applications for information on quality characteristics in the applications (see Figure 4).
6	Systematic Literature Review (SLR) [37]	Analysis of 50 articles on the theoretical idea, comparative study, and defined quality characteristics to gain input of issues, gaps, and points of discussion for selected quality characteristics. The input has formed the basis for designing a conceptual idea (see Figure 5).
7	Conceptual idea	A diagram that visualizes the theoretical idea with specific components for an educational application. Each detailed item was assigned a code to identify the proposed elements (see Figure 6).
8	Proposed checklist	Based on the conceptual diagram, a checklist (see Figure 7) was produced as a tool to guide software engineers, especially novices.

Fig. 2 represents the modified theoretical framework for the research originally adopted from D&M IS Success Model, as illustrated in the blue outlined rectangle [38]. The modification of the D&M IS Success Model presents the subject matter, which is quality characteristics, users, and discipline.

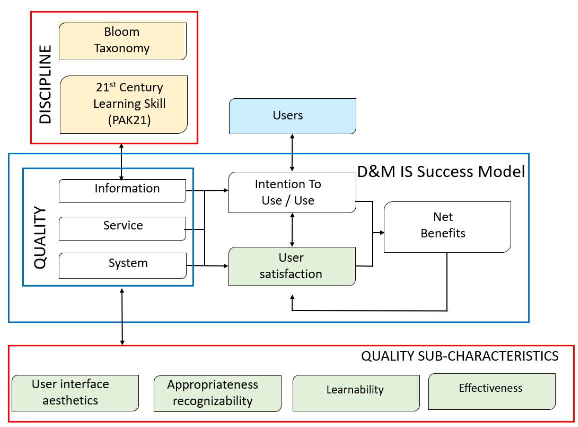

FIGURE 2. Theoretical framework.

From the theoretical framework, the study further defines and selects the quality sub-characteristics for the

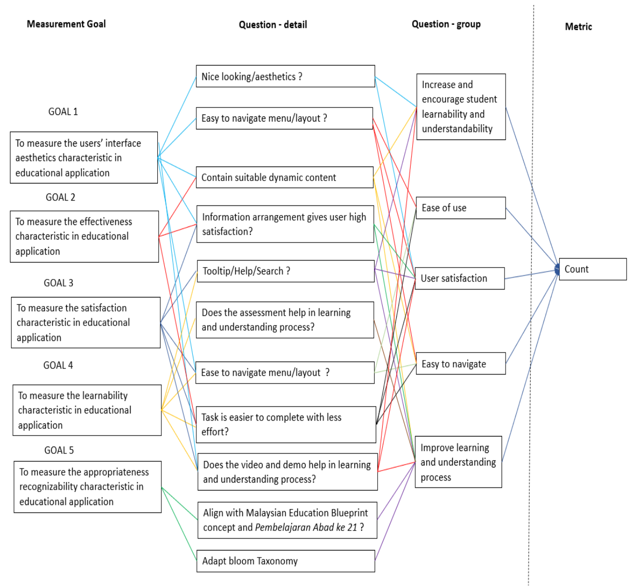

FIGURE 3. GQM quality metric indicator.

development of educational software applications. Fig. 3 represents the applied GQM that includes the measured goals, the questions used in gaining feedback regarding the quality, and the measured metrics.

The research quality characteristics which are user interface aesthetics, appropriateness recognizability, learnability, satisfaction, and effectiveness were selected based on the feedback and results of the GQM process. Fig. 4 represents the comparative study of the defined quality characteristics against existing online educational applications to rectify the quality features.

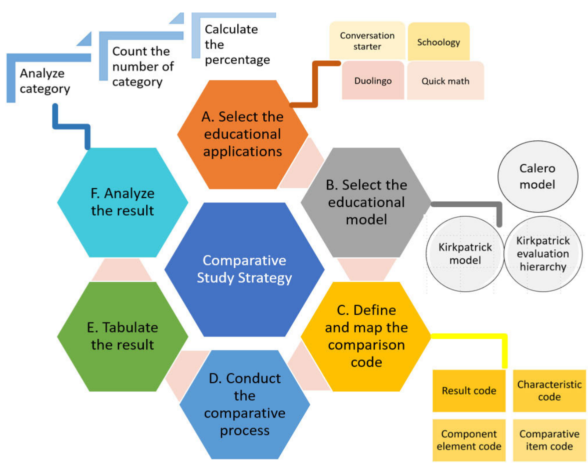

FIGURE 4. Comparative study process.

Fig. 5 shows the quality characteristics keywords used in the search step of the SLR process. The SLR process involves four steps: (1) preparing the research questions, (2) searching the process by using a search keyword, and (3) analyzing the process that involves the inclusion and exclusion

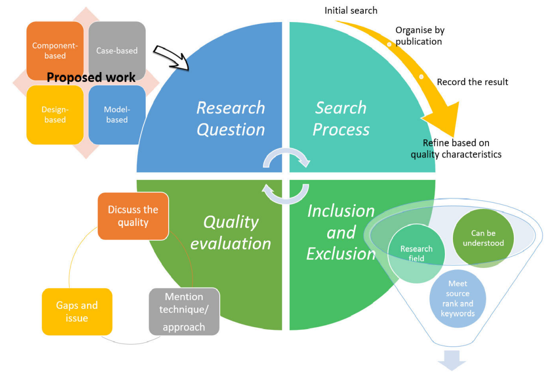

FIGURE 5. SLR process.

criteria. Based on the quality evaluation, only 50 papers were reviewed to ensure they met the research questions [39].

The selected articles that directly focus on each of the search strings in the SLR are listed in Table 5.

TABLE 5. Selected articles by search string.

Search String	Selected Articles	Total
User interface aesthetic	Zen et al., 2017 [40]; Gao, 2018 [41]; Nasruddin et al., 2018 [42]; Babic et al., 2018 [43]; Gogh & Kovacs, 2017 [44]; Mkpojiogu et al., 2018 [45]; Soui et al., 2022 [46]; Africa et al., 2015 [47]	8
Learnability	Prabhakaran et al., 2018 [48]; Zhang et al., 2017 [49]; Jonathan et al., 2016 [50]; Toda et al., 2015 [51]; Graf et al., 2015 [52]	5
Appropriateness recognizability	Finn, 2020 [53]; Kuo & Tseng, 2019 [54]; Eka & Retnani, 2018 [55]; Seman et al., 2016 [56]; Cheng et al., 2016 [57]	5
Satisfaction	O'Mahony et al., 2019 [58]; Sun & Li, 2017 [59]; Zhao et al., 2018 [60]; Liu & Zhao, 2018 [61]; Tsukamoto & Takemura, 2015 [62]; Schulz et al., 2015 [63]	6
Effectiveness	Schobel et al., 2018 [64]; Li et al., 2018 [61]; Baruah et al., 2017 [65]	3
Usability	Rosyidah et al., 2019 [66]; Arteaga and Rivera, 2019 [67]; Delgado et al., 2018 [68]; Al-Sumaty and Irfan, 2018 [69]; Putra et al., 2018 [70]; Yang et al., 2018 [71]; Alaa et al., 2018 [72]; Zhang et al., 2017 [49]; Meiland et al., 2017 [73]; Roldan et al., 2016 [74]; Zou et al., 2015 [75]	11
Technique	Chong et al., 2021 [76]; Meinema et al., 2019 [77]; Watts & Li, 2019 [78]; Chan et al., 2018 [79]; Ayed et al., 2017 [80]; Bennani et al., 2017 [81]; Bo et al., 2017 [82]; Prosvirin & Kharchenko, 2015 [83]; Ke et al., 2015 [84]; Berriche et al., 2015 [85]; Lu & Saniie, 2015 [86]; Gherardi & Hochgeschwender, 2015 [87]	12
	Grand Total	50

B. THE PROPOSED CHECKLIST APPROACH

The results from the SLR provide the input for the proposed framework known as CuE that adopts the checklist approach to guide the development of educational applications by NSDs (see Fig. 6). The checklist approach consists of three components in the FR that are user interface design [P1], learning [P2] and support [P3]. The guided FRs will adapt what should be in the NFRs in terms of software product quality and quality in use. In the concerned education domain, the proposed framework also aligns with the disciplines that include the Bloom Taxonomy and 21st century learning skills. Thus, NSDs should comply with the items in the checklist, execute the process and further evaluate the developed educational application.

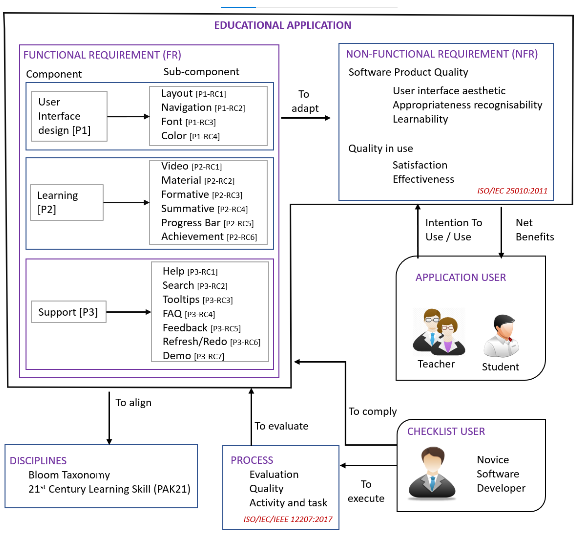

FIGURE 6. Proposed framework that includes the checklist approach as the guideline for the development of educational applications.

Fig. 7 shows the components and sub-components in the proposed framework that are mapped to the checklist. Information from activities (no. 1 – 8) as listed in Table 4 and diagrams (Fig. 2 to Fig. 6) have been used and referred to in producing the proposed checklist.

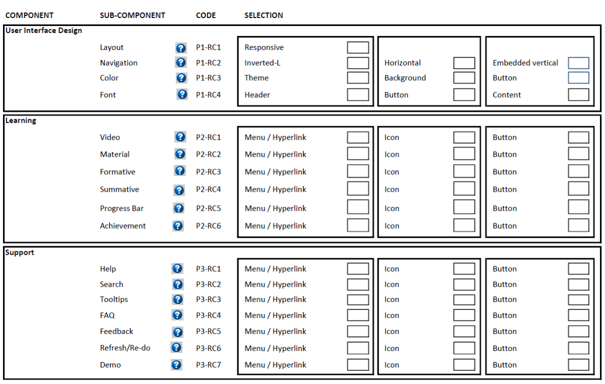

FIGURE 7. Proposed checklist.

In addition, this study also refers to existing educational models and guidelines, as listed in Table 6, for creating the proposed framework that includes the checklist approach.

TABLE 6. Educational guideline.

Model/ Discipline	Description
Kirkpatrick Model [88]	Mentions four logical levels in the educational process, namely, reaction, learning, behavior and results.
Kirkpatrick evaluation level [89]	Assigns the arrangement of the educational logical levels introduced by Kirkpatrick.
Malaysia Educational Blueprint 2013-2025 (MEB) [90]	Mentions the mission and vision of the Malaysian education strategy.
21st Century Learning Skills (PAK21) [91]	Promotes the creative approach in conducting a lesson with the active participation of students and teachers.
Bloom Taxonomy [92]	A guide to help teachers encourage students to develop essential critical thinking skills consist of six major categories: remembering, understanding, applying, analyzing, evaluating, and creating.

TABLE 7. Pre- and post-survey question on NFRs in components for educational applications.

Q-1. Are you aware that usability is a non-functional requirement applied in any software development process?
Q-2. Assign a score to the usability characteristics according to your knowledge and understanding.
Q-3. Are you aware that there are components that must be included in the development of educational application?
Q-4. Based on your knowledge and understanding, assign a score to a component or system function that should be included in the development of an educational application.
Q-5. Do you agree that a checklist can be used as a guideline in the development of an educational application?

Items of the same category are grouped accordingly in the checklist for ease of reference by the NSDs. The explanation based on the given codes is simple, easy to track and remember. The codes are (P1-RC1, P1-RC2, P1-RC3, P1-RC4, P2-RC1, P2-RC2, P2-RC3, P2-RC4, P2-RC5, P2-RC6, P3-RC1, P3-RC2, P3-RC3, P3-RC5, P3-RC6, and P3-RC7) as stated in Table 6 and tagged in Fig. 6.

The guidance process uses a coaching technique, where the leader or senior and experienced software developers become facilitators, and closely monitor the progress of the group project. Coaching instruments in previous research complement any advancement or diversion of the project [93]. For example, if the application needs to cater to desktop and smartphone devices, it must be adaptable to the responsive (P1-RC1) technique.

This approach promotes better flexibility and functionality of display on both devices and different screen sizes. As illustrated in Fig. 1, the creation of the checklist adopted the Agile Scrum Methodology process, which starts with a briefing during the requirements analysis session. In Sprint 1, before the start of the coaching session, 30 NSDs were requested to test their understanding and awareness of the NFRs by answering a short online survey. They were required to answer the five questions in Table 7, a pre-survey on NFRs and educational application components.

After completing the survey and attending the knowledge-sharing session, they were exposed to the NFRs needed for the educational application. The details in the checklist were explained to the NSDs as a guide for them in developing the educational application. Then, the NSDs were requested

TABLE 8. List of item codes and descriptions.

Component Code	Sub-component	Description	Quality characteristics				
			UA	*AR*	*LB*	*SF*	*EF*
P1-RC1	Layout	Responsive Website Design involves ensuring a website looks good on all devices such as desktops, tablets, and phones	√			√	√
P1-RC2	Navigation	Website navigation allows visitors to move from one page tothe other without frustration. It contains other convenient links and user interface elements within a website. Short and on-point navigation links also create an uncluttered appearance.			√	√	√
P1-RC3	Color	The right color is essential not only for aesthetics purposes. Colors promote a better user experience by simplifying navigation throughout the application and play a vital role in how the brain perceives the provided information and functionality.			√		√
P1-RC4	Font	Promotes readability and accuracy of learning by choosing fonts that do not appear to hug each other and improve spaces between letters.	√			√	
P2-RC1	Video	The video and audio features help students learn something that the teacher would be required to do otherwise. Video tutorials make the learning process fun and effective.		√	√		
P2-RC2	Material	Learning materials allow students to read the study materials from the education application with a few simple clicks. Students can also discover miscellaneous study materials by using education apps.	√	√	√		
P2-RC3	Formative	A process to check students' understanding and guide teachers' decision - making process for future instructions/plans.		√			√
P2-RC4	Summative	The final process in the learning cycle involves the assessment of students' skills and knowledge.		√			√
P2-RC5	Progress bar	A time-management tool for students to track progress and completion of activities.	√			√	√
P2-RC6	Achievement	Motivates students in the learning process by indicating their assessment/progress score. It helps teachers in monitoring the assessment results for further plans/activities.			√	√	√
P3-RC1	Help	Users get help in a separate window/page with the program's essential functions. It is part of a computer program that gives instructions and information about how to use the program.	√		√		√
P3-RC2	Search	Users can search for new terms/issues.	√	√	√		√
P3-RC3	Tooltips	Tooltips are user-triggered messages that provide additional information about a page element or feature.	√	√	√		√
P3-RC4	Frequently asked questions (FAQ)	Introduces newcomers to a topic or answers common questions in a document based on a question and answer format.		√	√		√
P3-RC5	Feedback	A medium that allows users to provide their feedback to other parties.	√			√	√
P3-RC6	Refresh/ Re-do	Refresh or re-do is a strategy used to restart an incomplete task rather than fixing it. It helps users to re-correct or reproduces their effort in the learning process. The refresh/reset button motivates users to expose their ideas without fear since they can refresh the page when they are not confident with their choices.				√	√
P3-RC7	Demo	Provides users with a quick guide on using the app that saves time and effort compared to manuals that impart extensive information.			√	√	√

Note : UA for user interface aesthetics; LB for learnability; AR for appropriateness recognizability; SF for satisfaction; EF for effectiveness

to answer the same survey questions (see Table 7) for the second time to test their understanding and awareness of NFRs. The purpose of this activity was to investigate whether their understanding had improved.

The checklist consists of three main components represented by the respective codes, P1 (user interface design) P2 (learning), and P3 (support) as detailed in Table 8.

The pre-and post-surveys comprise two types of questions, namely, Likert-style (Q-2, Q-4 and Q5) and multiple-choice questions (Q-1 and Q-3). The data of these surveys was recorded. The statistical technique evaluates the activities involved in the collection of research data [94]. To obtain analytic results, the Likert scale was mapped to the score marks, as displayed in Table 9. Two types of Likert scales were mapped to each score mark.

The analysis in Fig. 8 shows significant improvement in awareness among NSDs. In addition, the post-survey analysis indicates that all NSDs were aware that usability is a NFR that should be applied in any software development. This finding proves the briefing during the coaching sessions helped increase understanding and knowledge.

TABLE 9. Likert scale mapped to score marks.

Likert-Scale		Score Mark
Strongly Agree	Excellent	5
Agree	Good	4
Neutral	Fair	3
Disagree	Poor	2
Strongly Disagree	Very Poor	1

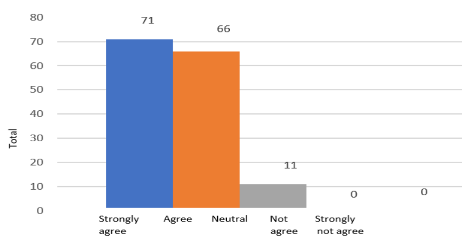

FIGURE 8. Q-1: Awareness among NSDs on non-functional requirements for inclusion in educational application.

After the briefing session, only three NSDs were unaware of the components to be included in the educational application (see Fig. 9) as compared to six NSDs before the

TABLE 10. Variance for Q2, Q4, and Q5.

Question	Pre-survey	Post-survey
Q2	35.170	43.000
Q4	32.530	52.760
Q5	29.500	48.500

briefing session. In the beginning, 24 NSDs were already aware of the components involved in educational applications, but after the briefing, three more NSDs gained awareness and better understanding.

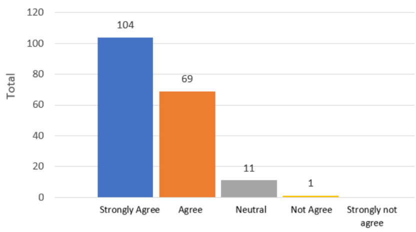

FIGURE 9. Q-3: Awareness among NSDs of components for inclusion in educational application.

Table 10 shows the variance values for questions Q-2, Q-4, and Q5 and Fig. 10 plots pre- and post-survey variance. The linear regression analysis determined the linear regression values. This analysis applies where the free variables X and Y are dichotomous or ceaseless [95].

Generally, the linear regression analysis involves testing a straight balance theory utilizing relapse examination by counting the items of X and Y that demonstrates the subordinate variable are widespread.

Linear regression calculates an equation that minimizes the distance between the fitted line and all the data points. R^2 is a statistical measure of the closeness of the data to the fitted regression line. Under a firm linear rule, values between 0.7 and 1.0 (-0.7 and -1.0) indicate a strong positive linear relationship [96].

This study defines the alpha value $= 0.5$ and confidence level $= 95\%$. The results show a strong positive linear relationship under a firm linear rule, where $R^2 = 0.9984$. It can be concluded that there is a significant relationship between the pre- and post-survey data.

The results for the Likert questions show the analytic results: (1) the correlation coefficient values to check the correlation between pre-survey and post-survey, and (2) and the Cronbach alpha values to test the reliability of the data [97], as listed in Tables 11 and 12.

This study investigated and calculated the correlation between Cronbach alpha values for reliability and regression for the dependent variables, as depicted in Table 11 (pre-survey) and Table 12 (post-survey), to prove significance and reliability. The Cronbach alpha value typically ranges from 0 to 1. Values closer to 1 indicate greater internal consistency of the variables on the scale. In other words, a higher Cronbach alpha value shows greater scale reliability.

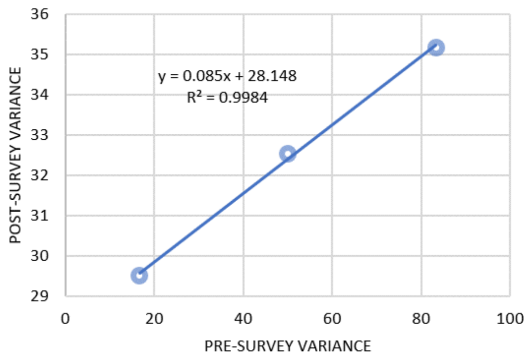

FIGURE 10. Linear regression graph for pre- and post-survey.

Cronbach alpha, also known as coefficient alpha, measures reliability, especially internal consistency, or item interrelatedness of a scale [98]. Internal consistency refers to the degree to which each item on the scale or test contributes positively towards measuring the equal assemble. Values in the direction of 1.0 imply an extra inner consistency of the variables on the scale. In different phrases, better Cronbach alpha values display higher scale reliability.

In this study, the results show a Cronbach alpha value of $\alpha = 0.8530$ in the pre-survey and $\alpha = 0.9075$ in the post-survey, and it can be concluded that the process has good scale reliability. The P-value was calculated for both surveys. This study defined the alpha value $= 0.5$, the degree of freedom df = N-2; 30-2 = 28 and confidence level = 95%. As stated in Table 11, the P-value is equal to 6.15417E-19 (P-value < 0.5), and in the post-survey (see Table 12), the P-value is equal to 1.47491E-30. It can be concluded that the P-value is <0.5 and there is a significant relationship between the pre-survey and post-survey results.

Table 13 shows the coefficient correlation between the pre-and post-survey Likert scale data. This study adopted the Pearson correlation coefficient method to measure the strength of the linear relationship [99] between each process.

The correlation value ranges from -1 to 1, where the value 1 represents a perfect positive correlation. A closer value to 1 indicates a stronger relationship between the two variables [100]. The findings in this study shows a strong correlation exists between the pre-and post-survey results (0.8714).

IV. IMPLEMENTATION OF THE CHECKLIST

As for the implementation, 37 NSDs referred to the checklist in the process of developing their educational software applications to fulfil a course under their Computer Science degree program, specializing in Software Engineering. Some of them enrolled in a Diploma program in Computer Science before pursuing their first degree. Therefore, the majority of them had little or no real-life experience in developing educational software.

TABLE 11. P-values and cronbach alpha for pre-survey.

Source of Variation	*SS*	*df*	*MS*	*F*	*P-value*	*F crit*	*Cronbach Alpha*
Rows	68.7398119	28	**2.45499328**	10.8072942	**1.47491E-30**	1.516659876	**0.907469901**
Columns	10.7586207	10	1.07586207	4.73612617	2.82228E-06	1.864604622	
Error	63.6050157	280	**0.22716077**				
Total	143.103448	318					

TABLE 12. P-value and cronbach alpha for post-survey.

Source of Variation	*SS*	*df*	*MS*	*F*	*P-value*	*F crit*	*Cronbach Alpha*
Rows	96.200627	28	**3.4357367**	6.8008155	**6.15417E-19**	1.516659876	**0.852958811**
Columns	15.090909	10	1.5090909	2.9871465	0.001350568	1.864604622	
Error	141.45455	280	**0.5051948**				
Total	252.74608	318					

TABLE 13. correlation coefficient value for pre-and post- survey.

	Pre-survey	**Post-survey**
Pre-survey	**1**	
Post-survey	**0.87140867**	**1**

They had a briefing session on the checklist. The NSDs had to create the assigned educational applications based on the given checklist. The NSDs were divided into eight teams to share their progress in each sprint meeting. The meetings took place every two weeks. Each team was given time to present their progress, which included the items that were listed in the checklist as part of user stories. The NSDs had to read the checklist carefully to confirm their understanding of both FRs and NFRs to avoid any ambiguity when referring to the checklist, especially quality characteristics needed in the educational applications.

In Sprint 2, they were required to share their progress and compliance with the updated checklist. The NSDs had to tick the selected items upon compliance, provide justification for non-compliance, or state any arguments. This study used a code for each compliance status, which is CL (comply), TB (to be done in next sprint), NS (status not stated), and EX (excluded from development). A total of 120 cells were checked for 15 sub-components.

Table 14 lists the feedback and findings gathered from all eight groups on compliance with the checklist. The results of the feedback were recorded and represented in a pie chart (see Fig. 11), and 5.83% indicated NS or did not state any indicator in their feedback cells, and 2.50% indicated EX or were excluded from using the checklist due to lack of expertise and time constraints.

Besides checking compliance with the checklist, this study distributed an online survey to gather more results on the effectiveness of the checklist. The online survey tested the NSDs' understanding, appropriateness recognizability, and satisfaction. The adaptation of UID led to the development of effective educational applications. Table 15 shows the questions in the survey questionnaire. Four questions (Q1 to Q4) were multiple questions with a Likert answer. All questions were designed to check the five quality sub-characteristics (understanding, appropriateness recognizability, satisfaction, user interface aesthetics, and effectiveness).

The value for Q1 to Q4 was calculated for a histography. The NSDs assigned their preferred score for Q1.a, Q1.b, Q1.c, and Q1.d as mentioned in Table 15. A total of 37 NSDs and 148 scores were collected. Fig. 12 shows the highest score scale, which is 71, chose 'Strongly agree' or 47.97% of the total questions answered. A total of 44.59% (66 out of 148) answered 'Agree', and 7.43% chose 'Neutral'. None of the NSDs chose 'Disagree' or 'Strongly Disagree'.

Fig. 13 shows the calculation for Q2(a) to Q2(e) with the highest score scale, 'Strongly agree' or 98 of the total questions answered at 52.97%. A total of 40.54% (75 out of 185) answered 'Agree'. While 6.49% chose 'Neutral', and none of the NSDs chose 'Disagree' or 'Strongly Disagree'. It can be concluded that most of the NSDs are aware that user interface aesthetics is a beneficial usability characteristic in the provided checklist.

While 6.49% chose 'Neutral', and none of the NSDs chose 'Disagree' or 'Strongly Disagree'. It can be concluded that most of the NSDs are aware that user interface aesthetics is a beneficial usability characteristic in the provided checklist

Fig. 14 shows the highest score scale, 104, answered 'Strongly agree', or 56.22% of the total questions answered. There are 37.30% (69 out of 185) who answered 'Agree'. A total of 5.95% answered 'Neutral', and only one or 0.54% chose 'Disagree'. None of the NSDs chose 'Strongly Disagree'. The findings show that NSDs have confirmed that the

TABLE 14. Feedback on checklist from novices.

Team	Sub-component															Number of compliance status			
	P1-RC1	P1-RC2	P1-RC3	P1-RC4	P1-RC5	P1-RC6	P2-RC7	P3-RC8	P3-RC9	P4-RC10	P5-RC11	P5-RC12	P5-RC13	P5-RC14	P5-RC15	CL	TB	NS	EX
T1	TB	TB	TB	TB	CL	TB	CL	TB	CL	CL	CL	CL	CL	CL	CL	**9**	**6**	**0**	**0**
	Justifications: 1. Sub-component P1-RC1, P1-RC2, P1-RC4, P1-RC6 & P3-RC8: 'To be define in next sprint' 2. Sub-component P1-RC3 : "The function not yet implement" 3. Sub-component P5-RC14 : 'Not necessary, we prefer to use video'																		
T2	NS	NS	TB	NS	CL	NS	CL	TB	TB	TB	CL	CL	CL	TB	TB	**5**	**6**	**4**	**0**
	Justifications: 1. Sub-component P5-RC14 P5-RC15, P1-RC3, P3-RC8 & P4-RC10: "Will be updated in the next sprint" 2. Sub-component P3-RC9 : "Will be updated but the UI is completed" 3. Sub-component P1-RC3 : "The function not yet implement" 4. Sub-component P5-RC14 : 'Not necessary, we prefer to use video'																		
T3	TB	TB	EX	CL	CL	TB	TB	CL	CL	TB	CL	CL	CL	TB	TB	7	7	**0**	**1**
	Justifications: 1. Sub-component P5-RC14 P5-RC15, P1-RC1, P3-RC2, P1-RC6 & P2-RC7: "Bring to next Sprint#3" 2. Sub-component P1-RC3 : "Not doing because at the nav bar already have subtopic" 3. Sub-component P3-RC8 : "For now I only state the score result for the assessment not included the feedback and reset"																		
T4	TB	TB	TB	CL	CL	TB	CL	TB	CL	TB	CL	CL	CL	TB	TB	**7**	**8**	**0**	**0**
	Justifications: 1. Sub-component P5-RC14, P5-RC15, P1-RC1 & P3-RC8: "Will be done in this sprint" 2. Sub-component P1-RC2 : "Will be done by person who is assigned with view performance" 3. Sub-component P1-RC6 : "Will be done by person who is assigned on use case manage user"																		
T5	TB	TB	EX	TB	CL	EX	CL	TB	TB	CL	CL	CL	CL	TB	CL	**7**	**6**	**0**	**2**
	Justifications: 1. Sub-component P1-RC1, P1-RC2, P1-RC4, P3-RC8 and P3-RC9: "Will comply for upcoming sprint" 2. Sub-component P5-RC14 : "We use video instead od audio because video is more interesting"																		
T6	NS	NS	CL	NS	CL	TB	CL	CL	CL	CL	CL	CL	CL	TB	CL	**10**	**2**	**3**	**0**
	Justifications: 1. Sub-component P1-RC6 & P5-RC14 : "Will comply for upcoming sprint"																		
T7	TB	CL	CL	CL	CL	TB	CL	TB	CL	TB	TB	TB	CL	CL	CL	**9**	**6**	**0**	**0**
	Justifications: 1. Sub-component P5-RC14 : "Currently still not implement any audio file" 2. Sub-component P1-RC6 : "Not implement it yet"																		
T8	CL	CL	CL	CL	CL	CL	CL	CL	CL	CL	CL	CL	CL	TB	CL	**14**	**1**	**0**	**0**
	Justifications: 1. Sub-component P5-RC14, P5-RC15, P1-RC1, P1-RC6 & P3-RC8: "Next sprint" 2. Sub-component P4-RC10 : "Nor complete yet"																		

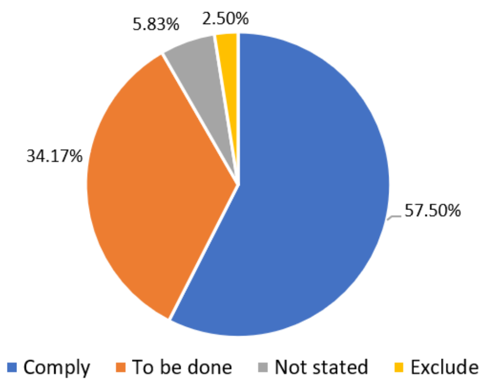

FIGURE 11. Data findings on checklist compliance.

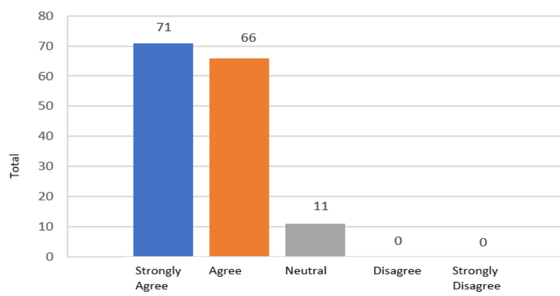

FIGURE 12. NSD Scores (Q1.a to Q1.d).

component and system functions that should be included in educational applications are learning materials, assessment, followed by feedback, video, and layout or navigation.

Fig. 15 shows the highest score scale, 178, answered 'Strongly agree' or 60.14% of the total answers. There are 34.12% (101 out of 296) who chose 'Agree'. A total of 5.74% chose 'Neutral'. Only for Q3, five questions were required to be answered by NSDs.

This study divided the answers into three categories, namely, (i) NSDs understand and benefit from the provided checklist; (ii) NSDs need further explanation; (iii) NSDs did not state anything.

The overall analysis in Fig. 16 shows that 89.19% of NSDs (34 out of 37) had understood and benefited from the provided checklist. A total of 8.11% (three NSDs) needed further explanation and only 2.70% (one NSD) did not give any feedback. This shows that most of them understood the needs listed in the proposed checklist.

V. THE EVALUATION

This section reports the evaluation of the educational applications developed by the NSD among the users. This study

TABLE 15. Survey questionnaire For NSDs.

Survey question	Sub-question
Q1. The statements that relate to educational applications	Q1.a: A checklist can be used as a guideline in application development Q1.b: Student understanding can be tested using formative and summative assessments Q1.c: A non-functional requirement should be applied in any software development process Q1.d: A good and attractive user interface design can reduce time in completing a given task
Q2. The usability characteristics that may benefit from the provided checklist	Q2.a: User Interface Aesthetics Q2.b: Appropriateness recognizability Q2.c: Learnability Q2.d: Satisfaction Q2.e: Effectiveness
Q3. The component and system function that should be included in educational applications	Q3.a: Learning materials Q3.b: Assessment (summative and formative) Q3.c: Feedback/Help/Tooltip/Search Q3.d: Audio & Video Q3.e: Standardization in layout/navigation/theme
Q4. The benefits of a dynamic and attractive content design	Q4.a: Increase and encourage learning ability Q4.b: Ease of use Q4.c: User satisfaction Q4.d: Ease of navigation Q4.e: Minimum supervision Q4.f: Reduce learning time Q4.g: Minimal action Q4.h: Improve learning process

TABLE 16. Likert scale answers by NSDs (Q1, Q2, Q3, and Q4).

Likert type 1	Likert type 2	Score	Q1	Q2	Q3	Q4
Strongly Agree	Excellent	5	71	98	104	178
Agree	Good	4	66	75	69	101
Neutral	Fair	3	11	12	11	17
Disagree	Poor	2	0	0	1	0
Strongly Disagree	Very poor	1	0	0	0	0

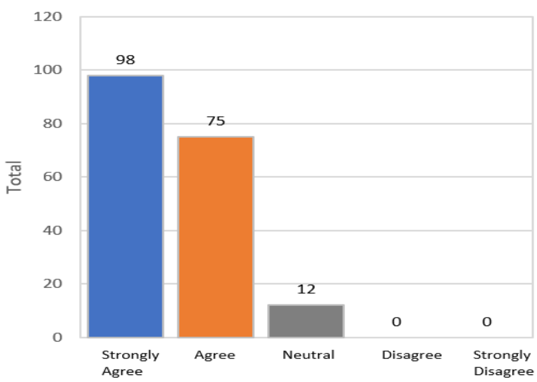

FIGURE 13. NSD Scores (Q2.a to Q2.e).

used two types of sampling methods. The first one was used to select the sample for teachers. The selected sample depends on purposive sampling or the judgment of the investigator, which is called subjective or judgment sampling. In this sampling method, the choice of the sample items depends exclusively on the judgment of the investigator [101], [102].

The second method is a Yamane formula [103]. It was used to decide how many samples are suitable for students [103], as mentioned in Section 3.2.3. We can assume the population

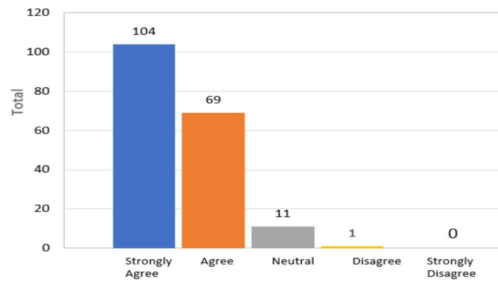

FIGURE 14. NSD Scores (Q3.a to Q3.e).

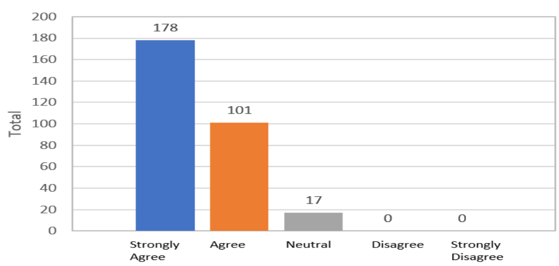

FIGURE 15. NSD Scores (Q4.a to Q4.h).

size and confidence level to find the acceptable sampling size [104], [105]. A total of 64 students from secondary school who enrolled Computer Science as their elective subject were the potential sample to become a population of this study. The sample size was calculated as follows:

$$n = \frac{\mathrm{N}}{1 + Ne^2}$$

where

n = sample size
N = the population size
e = the margin error in the calculation
*95% confident level and margin error = 0.5 are assumed

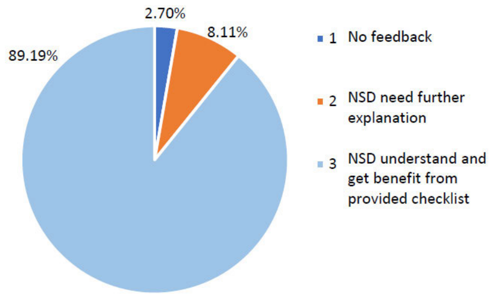

FIGURE 16. Categories of feedback by NSDs.

So,

$$
\begin{aligned}
n &= 64/(1+64(0.5)2)\\
&= 64/17\\
&= 3.76\\
&= 4
\end{aligned}
$$

Therefore, the minimum sample size for this study is four. A total of 42 students participated. Next, for teachers and senior software developers, this study applied non-random sampling where purposive sampling or judgement sampling was chosen. The purposive sampling focused only on the target respondents, and this study managed to obtain responses from five teachers and 16 senior software developers. They provided feedback by answering the structured interview questions after they accessed and used the WALCS application. Next, purposive sampling was also used for the involvement of five experts in assessing the proposed framework that adopts the checklist approach using the prototype tool towards WALCS.

The subject selection adopted the Yamane sample size formula [103]. The feedback from 42 secondary level students from a school in Malaysia were recorded. The students accessed the application that implemented the proposed checklist and answered the survey and completed the given task. The questionnaire is shown in Table 17. Generally, the questions are related to the quality, user interface aesthetics, appropriateness recognizability, satisfaction, and effectiveness of the educational application.

Data from the survey was collected and analyzed. Fig. 17(a) shows that 24 (57%) of the respondents have their own computers at home, while Fig. 17(b) shows that only 2 (5%) of the respondents never had any experience using an educational application Quality questions (Q8) consists of eight sub-questions. The questions applied a Likert scale format from ''strongly agree'' to ''strongly disagree'' which represents the score marks from 5 to 1. Fig. 18 shows the highest value is for question Q8.g, where 26 out of 42 respondents agreed that quality helps in reducing repetitive action in completing a given task. The students believed that this characteristic is beneficial in motivating the learning activities that increases their satisfaction. Thus, the user interface aesthetics and effectiveness of the educational application increases their learning skills and understanding.

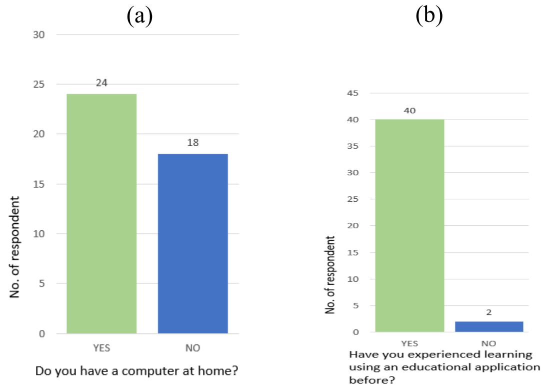

FIGURE 17. Background of respondents.

It can be concluded that 42 respondents have given a positive response to a quality question and the results show that they are aware of the benefits of quality characteristics in an educational application.

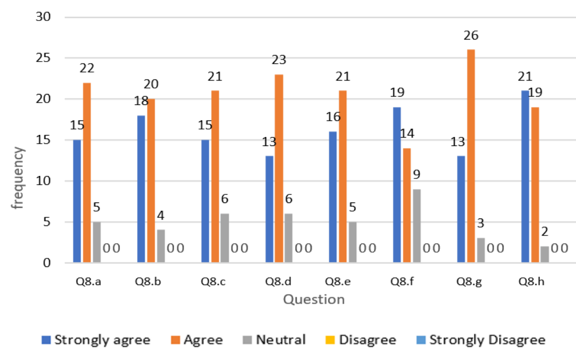

FIGURE 18. Relres.

As for user interface aesthetics, (Q3), consists of five sub-questions. Fig. 19 shows the highest value for question Q3.e, where 29 out of 42 respondents agreed that a vital factor that supports a good UID is the arrangement of the content in educational applications. Furthermore, the students also gave the highest score to the interface conditions such as nice color and ease of access. It can be concluded that software developers should specifically consider the user interface aesthetics in educational applications, especially its appearance to motivate users to explore more functions in the application.

The learnability aspect, (Q4), consists of five sub-questions. Fig. 20 shows the highest value for questions, Q4.a, (easy to access to learning materials), Q4.b is where the assessment was provided, and Q4.d is the FAQ. Out of 42 respondents, 25 respondents agreed that all these factors should be included in educational applications. It can be concluded that software developers should specifically consider the learnability sub-characteristic in the development of educational applications.

The question on appropriateness recognizability, (Q5), consists of four sub-questions. Based on the percentage value, most respondents (151 or 89.88%) showed a positive answer

TABLE 17. Questionnaire for students.

Question	Description
Q1	(a) Do you have a computer at home?
Q2	(b) Have you experienced learning using an educational application before?
Q3	The statements that relate to the design of the WALCS application
Q3.a.	The overall design looks good.
Q3.b.	The font is easy to see.
Q3.c.	The color is nice.
Q3.d.	The menu is easy to access.
Q3.e.	The arrangement of the application function is in a good manner.
Q4	The statements that relate to the learning process in using the WALCS application
Q4.a	Learning material is easy to access and properly arranged according to course syllabus.
Q4.b	*Praktis Kendiri* and quiz are good to test students' performance after finishing a topic.
Q4.c	*Panduan* is useful and helpful to guide students on how to use the WALCS application and learn about WALCS.
Q4.d	Frequently Answered Question (FAQ) is useful and helpful to gain information.
Q5	The statements that relate to the students understanding in using the WALCS application
Q5.a	A learning video can increase students' understanding.
Q5.b	Students were given 3 times to re-do the Quiz to increase their level of understanding.
Q5.c	Tooltips is a label note that helps students' understanding in accessing WALCS.
Q5.d	Help function increases students' understanding in accessing WALCS.
Q6	The statements that relate to the students' satisfaction in using the WALCS application
Q6.a	A progress bar in *Praktis Kendiri* and Quiz can help students manage their time when answering the question.
Q6.b	The feedback function allows student to share their queries/statements with teachers.
Q6.c	The search function helps to decrease effort and time in finding information to increase students' understanding.
Q6.d	The mark/achievement displayed after students answered *Praktis Kendiri* or Quiz helps to increase students' motivation in the learning activity.
Q6.e	The combination of audio and video is satisfying.
Q6.f	The download function for notes is helpful and satisfying.
Q7	The statements that relate to the students' effectiveness in using the WALCS application. Based on the students experience, give a score for each of the provided sub-component
Q7.a	Demo (a *Panduan* video)
Q7.b	Learning material
Q7.c	Formative assessment (Quiz)
Q7.d	Summative assessment (*Praktis Kendiri*)
Q7.e	Tooltip function (a short description for icon label)
Q8.	From students' point of view, the factors that benefit of learning activities from the implementation of dynamic WALCS and 3-looking content design, give a score for each of the factors
Q8.a	Increase and encourage learning ability
Q8.b	Easy to use
Q8.c	Increase user satisfaction
Q8.d	Ease of navigation
Q8.e	Minimum supervision
Q8.f	Reduce learning time and effort
Q8.g	Minimal action to complete given task
Q8.h	Improve learning process
Q9	Overall feedback on the usefulness of the WALCS in learning activities
Q10	Suggestion for improvement

(either 'strongly agree' or 'agree'). It can be concluded that educational applications should be developed by complying with the proposed checklist such as providing video and formative assessment functions. Thus, tooltips and help functions are under the support components that manage extra information and guidance to the users. Fig. 21 shows the highest value for questions Q5.b, where 26 out of 42 respondents agreed that students should be given more than once chance to repeat their formative assessment to improve their level of understanding.

In addition, satisfaction (Q6) consists of six sub-questions. Based on the percentage value, the majority of respondents (231 or 91.68%) showed a positive answer (either 'strongly agree' or 'agree'). Fig. 22 shows the highest value for questions Q6.a, where 26 out of 42 respondents agreed that the application should provide a progress bar status to help students manage time in answering questions. It can be concluded that software developers should specifically consider the satisfaction characteristic when developing educational applications.

The effectiveness characteristic (Q7) consists of five sub-questions. The percentage shows that 182 (89.67%) gave a positive answer, (either 'strongly agree' or 'agree'). Only 1.43% or three respondents provided a negative answer. The formative and summative assessments allow students to test their skills and understanding level. Additional help from the tooltip feature also prevents wrong assumptions.

TABLE 18. Qualitative comparison of CuE with existing frameworks.

Work related to software quality	Code Review Checklist [76]	Criterion checklist [106]	Quality checklist [22]	Checklists for evaluations [78]	Methodological quality checklist [107]	CIPP evaluation checklist [108]	CuE checklist
Quality characteristic							
User interface aesthetic	X	X	X	X	X	X	√
Appropriateness recognizability	√	√	√	√	√	√	√
Learnability	√	X	√	√	X	√	√
Satisfaction	X	√	√	√	X	√	√
Effectiveness	√	X	X	X	√	√	√

Work related to checklist	Issue	Improvement by CuE
Code Review Checklist [76]	Users had misconceptions about the purpose of code review	CuE provided a detailed explanation including the purpose for each sub-component
Criterion checklist [106]	Fail to follow every criterion because it has various criteria across multiple checklists.	The guideline only in one checklist
Quality checklist [22]	The checklist provided a list of criteria or instructions that respondents need to check and confirm; do not provide a sample.	CuE provided a tool to visualize a guidance and also a sample for each of the sub-components
Checklists for evaluations [78]	Issue with subjective nature of checklists and not being comprehensive enough to help users	CuE provided a tool to visualize a guidance
Methodological quality checklist [107]	When executing this type of activity, novices and professionals who were not experts in the methods did not have a basic foundation to guide their activities	CuE is straight forward and is in simple notation and explanation; no special skills are needed to use it.
CIPP evaluation checklist [108]	The concepts were taught in a logical and complex manner to be fully implemented	CuE has represented a logical diagram into a class diagram, activity diagram and a CuE to simplify users' understanding.

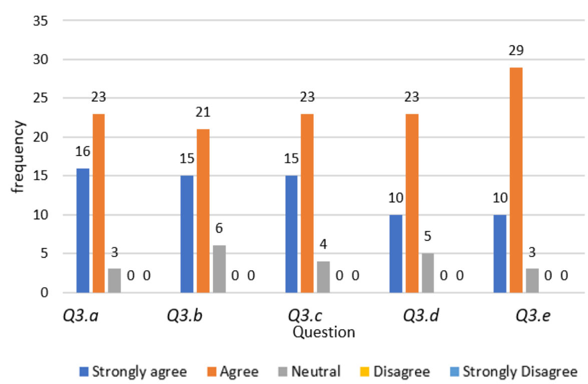

FIGURE 19. **User interface aesthetics analysis.**

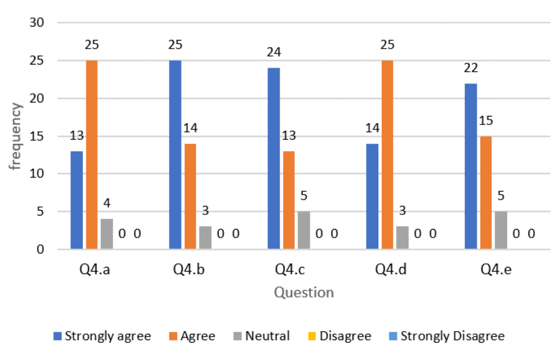

FIGURE 20. **Learnability analysis.**

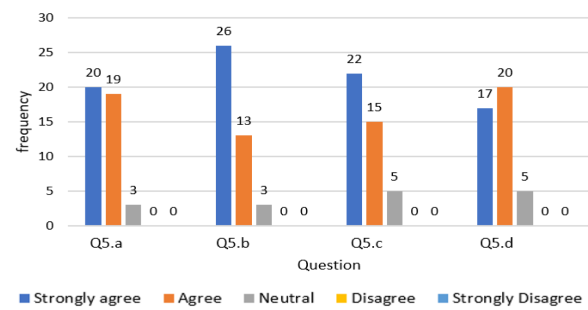

FIGURE 21. **Appropriateness recognizability analysis.**

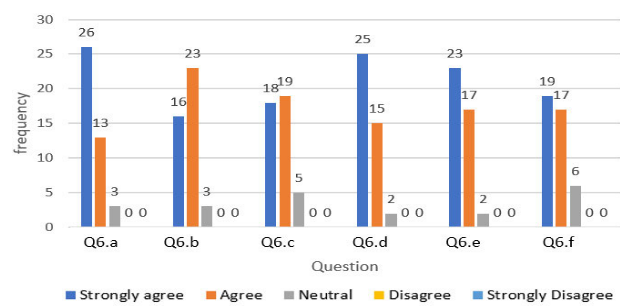

FIGURE 22. **Satisfaction analysis.**

The works related to software quality and checklist were independently compared to evaluate the proposed work. User interface aesthetic, appropriateness recognizability, learnability, and effectiveness sub-characteristics, besides the sub-components and requirements are listed in Table 18. Recent works were compared with the proposed checklist approach in the CuE framework that provides some improvements. The table shows that the CuE framework with the

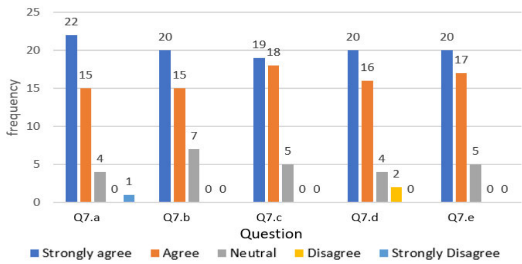

FIGURE 23. Effectiveness analysis.

checklist approach covers all the quality sub-characteristics with some improvements in related issues as further elaborated in the following paragraphs.

None of the existing works mentioned user interface aesthetics, except the proposed CuE with checklist. Thus, only the CuE framework included all five quality sub-characteristics. Meanwhile, other frameworks just included between two to four sub-characteristics. Besides the five quality sub-characteristics, the comparison also takes into consideration the issues related to checklists. From the comparative table, the CuE manages to provide guidance for NSDs and at the same time minimizes or overcomes any stated issues.

In the effort to avoid NSDs misconceptions, the CuE with the checklist approach provides a detailed explanation when NSDs click on the question mark icon in the checklist prototype tool. Thus, to help NSDs determine the selected factors, the CuE provides an example in picture format to give a clear understanding for developers. Indeed, the checklist is comprehensive enough to help users, while the CuE framework visualizes the required information as a guidance. Therefore, this research shows that the CuE with the checklist approach can fill the gaps in existing works that guide NSDs when developing educational applications.

VI. CONCLUSION AND FUTURE WORK

The study concludes that the proposed checklist approach could be the main reference in developing educational software applications. The findings deduce that the proposed checklist helps NSDs ensure the incorporation of elements and components that are essential in educational applications. By implementing the checklist approach, a team leader can monitor the development process systematically and make sure educational applications align with the stakeholders' requirements in the education industry.

In creating the checklist, the analysis obtained from the pre-and post-study on awareness among NSDs on usability shows a linear regression graph exists. The correlation coefficient value is strong (0.8714), and the Cronbach alpha value to measure the internal consistency of the variables on the scale is close to 1 (perfect consistency). Good Cronbach alpha values display better reliability. In this study, the results of the pre-survey (0.8529), and post-survey (0.9075) prove that the process has a high reliability.

In the checklist implementation activities, a survey was conducted, and the results showed that in Sprint 2, the NSDs shared their progress of 56.6% compliance with the items in the proposed checklist for the development of educational applications.

The correlation coefficient value is strong, where the value is >0.9. In addition, the Cronbach alpha that measures the internal consistency of the variables on the scale is close to 1 (perfect consistency). The results for the pre-survey showed a value of 0.9267. As for effectiveness, the findings showed that 89.19% of NSDs (34 out of 37) understood and benefited from the provided checklist in creating educational applications.

Based on the execution of the CuE framework with the checklist approach on a real-world application (WALCS), several contributions are worthy of indication in two aspects, which are theoretical contribution and practical contribution.

A. THEORETICAL CONTRIBUTION

The SLR technique employed in this work can provide systematic information and insights to other researchers. The empirical part of the evaluation includes five sets of hypotheses, and the proposed CuE with checklist can support the necessity for user interface aesthetic, appropriateness recognizability, learnability, satisfaction, and effectiveness variables to be considered in educational applications. These characteristics and sub characteristics are enhancement characteristics in the Delone & Mclean Model [38].

The correlation between all five defined quality characteristics and sub-characteristics of this study shows that both data sets have a strong association, allowing us to decide whether the data values are "good enough" fit for all five hypotheses. The proven five sets of hypotheses show that the decision on selecting the five defined quality characteristics and sub-characteristic is suitable enough to be included in the CuE.

Theoretically, the CuE is a diagram that illustrates the proposed work where software developers, especially NSDs, can conceptually refer to the components, sub-components, elements and the relationship between each connection. Thus, the composite structure diagrams and activity diagrams are another representation of the CuE that NSDs can refer to. All the diagrams represent both technical and theoretical views to NSDs. Hence, researchers and NSDs can refer to various information to understand further the proposed CuE with checklist approach.

B. PRACTICAL CONTRIBUTION

Practically, the CuE framework that includes the checklist approach in a checklist format as in the CuE prototype tool can provide guidance for NSDs in gaining a common understanding of UID criteria in a straightforward and consistent manner. The prototype tool allows NSDs to explore the guideline using any web browsers. NSDS can refer to the CuE at any time. From the tool, the CuE framework can also be printed and distributed to others.

Typically, developers always have their own knowledge, definitions and understanding about application development, and NSDs must have a clear understanding to avoid any issues, especially in the requirements and development phase. Misunderstanding can create conflict that leads to difficult situations, especially when they work in a group and do not have any guidance to refer to. However, by referring to the CuE, they can increase their understanding in the right way.

Additionally, the CuE framework can assist NSDs to learn what components should be included in educational applications, as well as any other requirements that users may request. Thus, The CuE framework can assist in the standardization and consistency of user needs. Users' requirements change regularly depending on their needs, but with the CuE design document generated by the CuE application, they can return to the earlier criteria for the UID if a dispute arises. Since it is not required for any specific purpose and can be accessed via any type of Web browser, the CuE framework can be utilized by both novices and expert developers in educational application development.

Hence, future studies could explore further how the proposed checklist approach can be part of a compressive framework for the development of educational applications by NSDs. Thus, the CuE framework can be extended and improved in multiple ways for future research as follows:

(i) Include more quality sub-characteristics, such as completeness and efficiency. Completeness and efficiency are thought to be related to user interface aesthetics, appropriateness, recognizability, learnability, satisfaction, and effectiveness.
(ii) Consider more sub-components in the future to cover a variety of quality sub-characteristics, such as maintainability, which can help to optimize system configuration and provide a personalized dashboard.
(iii) Insert more handicapped-friendly requirements such as the voice guidance in the CuE framework.
(iv) Run a controlled experiment with several groups of people, including parents.
(v) Utilizing other instruments in future experiments that may capture varied information, such as eyeball tracking to learn about respondents' habits when using the CuE framework.

Finally, future research may also explore whether the checklist approach in the CuE framework can be customized and benefited for different application domains.

ACKNOWLEDGMENT

The authors would like to thank Dr. Jin Song Dong of the National University of Singapore for his constructive feedback on this study.

REFERENCES

[1] L. T. Burner, "Why is educational change so difficult and how can we make it more effective?" *Forskning Forandring*, vol. 1, no. 1, pp. 122–134, 2018.
[2] J. C. Garcia-Huidobro, A. Nannemann, C. K. Bacon, and K. Thompson, "Evolution in educational change: A literature review of the historical core of the journal of educational change," *J. Educ. Change*, vol. 18, no. 3, pp. 263–293, Aug. 2017.
[3] A. Ibrahim, "The application of educational software in teaching and learning for living skills in secondary schools," in *Proc. 1st Int. Teacher Educ. Conf. Teach. Pract.*, 2016, pp. 1–6.
[4] S. Moyo and E. Mnkandla, "A metasynthesis of solo software development methodologies," in *Proc. Int. Multidisciplinary Inf. Technol. Eng. Conf. (IMITEC)*, Vanderbijlpark, South Africa, Nov. 2019, pp. 1–8.
[5] S. Nan, P. V. Gorp, H. H. M. Korsten, U. Kaymak, R. Dovjak, X. Lu, and H. Duan, "DCCSS: A meta-model for dynamic clinical checklist support systems," in *Proc. 3rd Int. Conf. Model-Driven Eng. Softw. Develop. (MODELSWARD)*, 2015, pp. 272–279.
[6] C. Papoutsi, A. Poots, J. Clements, Z. Wyrko, N. Offord, and J. E. Reed, "Improving patient safety for older people in acute admissions: Implementation of the frailsafe checklist in 12 hospitals across the U.K.," *Age Ageing*, vol. 47, no. 2, pp. 311–317, Mar. 2018.
[7] J. S. Armstrong and K. C. Green, "Guidelines for science: Evidence and checklists," *SSRN Electron. J.*, pp. 1–37, Oct. 2018.
[8] B. Hales, M. Terblanche, R. Fowler, and W. Sibbald, "Development of medical checklists for improved quality of patient care," *Int. J. Quality Health Care*, vol. 20, no. 1, pp. 22–30, Nov. 2007.
[9] J. Wen, Z. Chen, Y. Liu, Y. Lou, Y. Ma, G. Huang, X. Jin, and X. Liu, "An empirical study on challenges of application development in serverless computing," in *Proc. 29th ACM Joint Meeting Eur. Softw. Eng. Conf. Symp. Found. Softw. Eng.*, Aug. 2021, pp. 416–428.
[10] D. Votipka, D. Abrokwa, and M. L. Mazurek, "Building and validating a scale for secure software development self-efficacy," in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, Apr. 2020, pp. 1–16.
[11] M. Alsaadi, A. C. Lisitsa, and M. Qasaimeh, "Minimizing the ambiguities in medical devices regulations based on software requirement engineering techniques," in *Proc. Int. Conf. Sci., E-Learn. Inf. Syst.*, 2019, pp. 1–4.
[12] D. Sabella, A. Alleman, E. Liao, M. Filippou, Z. Ding, and L. G. Baltar, "Edge computing: From standard to actual infrsastructure deployment and software development," ETSI, Sophia Antipolis, France, Tech. Rep., 2019, pp. 1–41.
[13] S. Oyedeji, B. Penzenstadler, M. O. Adisa, and A. Wolf, "Validation study of a framework for sustainable software system design and development," in *Proc. CEUR Workshop*, 2019, p. 2382.
[14] AB Testing. (Feb. 5, 2022). *Process.st*. [Online]. Available: https://www.process.st/checklist/ab-testing/Abbas
[15] S. M. Abbas, K. A. Alam, U. Iqbal, and S. Ajmal, "Quality factors enhancement of requirement engineering: A systematic literature review," in *Proc. Int. Conf. Frontiers Inf. Technol. (FIT)*, Dec. 2019, pp. 13–18.
[16] C. A. Cortes-Camarillo, V. Y. Rosales-Morales, L. N. Sanchez-Morales, G. Alor-Hernandez, and L. Rodriguez-Mazahua, "Atila: A UIDPs-based educational application generator for mobile devices," in *Proc. Int. Conf. Electron., Commun. Comput. (CONIELECOMP)*, 2017, pp. 1–7.
[17] H. Shariff, "Non-functional requirement detection using machine learning and natural language processing," *Turkish J. Comput. Math. Educ.*, vol. 12, no. 3, pp. 2224–2229, Apr. 2021.
[18] L. Huang and K.-S. Ma, "Introducing machine learning to first-year undergraduate engineering students through an authentic and active learning labware," in *Proc. IEEE Frontiers Educ. Conf. (FIE)*, Oct. 2018, pp. 1–4.
[19] S. Zhou and Y. Zhang, "Active learning for cost-sensitive classification using logistic regression model," in *Proc. IEEE Int. Conf. Big Data Anal. (ICBDA)*, Mar. 2016, pp. 1–4.
[20] S. Wicha, K. Tangmongkhonnam, K. Saelim, T. Jongkorklang, and K. Khant, "The developing of active English learning system for local entrepreneurs," in *Proc. Int. Conf. Digit. Arts, Media Technol. (ICDAMT)*, 2017, pp. 183–188.
[21] A. Chaparro, J. R. Keebler, E. H. Lazzara, and A. Diamond, "Checklists: A review of their origins, benefits, and current uses as a cognitive aid in medicine," *Ergonom. Des., Quart. Hum. Factors Appl.*, vol. 27, no. 2, pp. 21–26, Apr. 2019.
[22] J. S. Molléri, K. Petersen, and E. Mendes, "An empirically evaluated checklist for surveys in software engineering," *Inf. Softw. Technol.*, vol. 119, Mar. 2020, Art. no. 106240.
[23] *Environmental Management Systems—Quirements With Guidance for Use*, ISO Standard 14001:2015, 2015.
[24] *Quality Management Systems—Requirements*, ISO Standard 9001:2015, 2015.
[25] *Quality Management—Quality of an Organization Guidance to Achieve Sustained Success*, ISO Standard 9004:2018, 2018.

[26] *Occupational Health and Safety Management Systems—Requirements With Guidance for Use*, ISO Standard 45001:2018, 2018.
[27] *Food Safety Management Systems—Requirements for Any Organization in the Food Chain*, ISO Standard 22000:2018, 2018.
[28] *General Requirements for the Competence of Testing and Calibration Laboratories*, ISO Standard 17025:2017, 2017.
[29] *Information Technology—Security Techniques—Information Security Management Systems—Requirements*, ISO Standard 27001:2013, 2013.
[30] *Medical Devices—Quality Management Systems—Requirements for Regulatory Purposes*, ISO Standard 13485:2016, 2016.
[31] K. A. Saed, N. Aziz, S. Jadid, and N. Hafizah, "Data governance cloud security checklist at infrastructure as a service (IaaS)," *Int. J. Adv. Comput. Sci. Appl.*, vol. 9, no. 10, pp. 1–10, 2018.
[32] W. Y. Higgins and D. J. Boorman, "An analysis of the effectiveness of checklists when combined with other processes, methods and tools to reduce risk in high hazard activities," *Boeing Tech. J.*, 2016.
[33] J. E. Akinsola, A. S. Ogunbanwo, O. J. Okesola, L. J. Odun-Ayo, F. D. Ayegbusi, and A. A. Adebiyi, "Comparative analysis of software development life cycle models (SDLC)," in *Proc. Comput. Sci. Conf.* Cham, Switzerland: Springer, 2020, pp. 310–322.
[34] (Nov. 10, 2022). *What is Scrum?*. [Online]. Available: https://www.scrum.org/resources/what-is-scrum
[35] A. López-Alcarria, A. Olivares-Vicente, and F. Poza-Vilches, "A systematic review of the use of agile methodologies in education to foster sustainability competencies," *Sustainability*, vol. 11, no. 10, p. 2915, May 2019.
[36] V. Basili, A. Trendowicz, M. Kowalczyk, J. Heidrich, C. Seaman, J. Münch, and D. Rombach, *Aligning Organizations Through Measurement The GQM+Strategies Approach*. Berlin, Germany: Springer, 2014, p. 205.
[37] B. Kitchenham, O. P. Brereton, D. Budgen, M. Turner, J. Bailey, and S. Linkman, "Systematic literature reviews in software engineering—A systematic literature review," *Inf. Softw. Technol.*, vol. 51, no. 1, pp. 7–15, 2009.
[38] W. Delone and E. Mclean, "The DeLone and McLean model of information systems success? A ten-year update," *J. Manage. Inf. Syst.*, vol. 19, no. 4, pp. 9–30, 2003.
[39] N. Ngadiman, S. Sulaiman, N. Idris, M. R. Samingan, and H. Mohamed, "Systematic review on software quality in educational applications," *IEEE Access*, vol. 9, pp. 60187–60200, 2021.
[40] M. Zen and J. Vanderdonckt, "Assessing user interface aesthetics based on the inter-subjectivity of judgment," in *Proc. Electron. Workshop Comput.*, Jul. 2016, pp. 1–12.
[41] Q. Gao, "The teaching reform of animation specialty in universities under the aesthetic education environment," in *Proc. 3rd Int. Conf. Politics, Econ. Law*, 2018, pp. 137–140.
[42] Z. A. Nasruddin, A. Markom, and M. A. Aziz, "Evaluating construction defect mobile app using think aloud," in *Proc. Int. Conf. User Sci. Eng.*, 2018, pp. 3–11.
[43] S. Babic, T. Orehovacki, and D. Etinger, "Perceived user experience and performance of intelligent personal assistants employed in higher education settings," in *Proc. 41st Int. Conv. Inf. Commun. Technol., Electron. Microelectron. (MIPRO)*, May 2018, pp. 830–834.
[44] E. Gogh, A. Kovacs, and G. Sziladi, "Application of E-diary to analyze the effectiveness of learning," in *Proc. 8th IEEE Int. Conf. Cognit. Infocommun. (CogInfoCom)*, Sep. 2017, pp. 419–424.
[45] E. O. C. Mkpojiogu, N. L. Hashim, and R. Adamu, "Observed demographic differentials in user perceived satisfaction on the usability of mobile banking applications," in *Proc. Knowl. Manage. Int. Conf.*, 2016, pp. 29–30.
[46] M. Soui, M. Chouchane, N. Bessghaier, M. W. Mkaouer, and M. Kessentini, "On the impact of aesthetic defects on the maintainability of mobile graphical user interfaces: An empirical study," *Inf. Syst. Frontiers*, vol. 24, no. 2, pp. 659–676, Apr. 2022.
[47] J. Ramdhani and S. Ramsaroop, "The reawakening of teaching through aesthetics in education: Students' perspectives," *Koers-Bull. Christian Scholarship*, vol. 80, no. 2, pp. 1–7, Dec. 2015.
[48] M. Prabhakaran, C. Pantina, G. Gutjahr, R. Raman, and P. Nedungadi, "Effectiveness of online labs teacher training workshop," in *Proc. IEEE 18th Int. Conf. Adv. Learn. Technol. (ICALT)*, Jul. 2018, pp. 249–251.
[49] J. Zhang, H. Xie, and H. Li, "Competency-based knowledge integration of BIM capstone in construction engineering and management education," *Int. J. Eng. Educ.*, vol. 33, no. 6, pp. 2020–2032, 2017.
[50] F. C. Jonathan, O. Karnalim, and M. Ayub, "Extending the effectiveness of algorithm visualization with performance comparison through evaluation-integrated development," in *Proc. Seminar Nasional Aplikasi Teknologi Informasi (SNATI)*, 2016, pp. 1–10.
[51] A. M. Toda, R. S. D. Carmo, V. Campos, A. L. da Silva, and J. D. Brancher, "Evaluation of SiGMa, an empiric study with math teachers," in *Proc. IEEE Frontiers Educ. Conf. (FIE)*, Oct. 2015, pp. 1–6.
[52] H. Graf, J. Keil, and A. Pagano, "A contextualized educational museum experience connecting objects, places and themes through mobile virtual museums," in *Proc. Digital Heritage*, Sep. 2015, pp. 337–340.
[53] B. Finn, "Exploring interactions between motivation and cognition to better shape self-regulated learning," *J. Appl. Res. Memory Cognition*, vol. 9, no. 4, pp. 461–467, 2020.
[54] H.-C. Kuo, Y.-C. Tseng, and Y.-T.-C. Yang, "Promoting college student's learning motivation and creativity through a STEM interdisciplinary PBL human–computer interaction system design and development course," *Thinking Skills Creativity*, vol. 31, pp. 1–10, Mar. 2019.
[55] W. E. Y. Retnani, B. Prasetyo, Y. P. Prayogi, M. A. Nizar, and R. M. Abdul, "Usability testing to evaluate the library's academic web site," in *Proc. 4th Int. Conf. Comput. Appl. Inf. Process. Technol. (CAIPT)*, Aug. 2017, pp. 3–6.
[56] L. O. Seman, G. Gomes, and R. Hausmann, "Statistical analysis using PLS of a project-based learning application in electrical engineering," *IEEE Latin Amer. Trans.*, vol. 14, no. 2, pp. 646–651, Feb. 2016.
[57] P.-H. Cheng, Y.-T. C. Yang, S.-H. G. Chang, and F.-R. R. Kuo, "5E mobile inquiry learning approach for enhancing learning motivation and scientific inquiry ability of university students," *IEEE Trans. Educ.*, vol. 59, no. 2, pp. 147–153, May 2016.
[58] N. O'Mahony, S. Campbell, A. Carvalho, S. Harapanahalli, V. G. Hernandez, and L. Krpalkova, "Deep learning vs. traditional computer vision," in *Proc. Sci. Inf. Conf.*, 2019, pp. 128–144.
[59] Y. Sun and Q. Li, "The application of 3D printing in mathematics education," in *Proc. 12th Int. Conf. Comput. Sci. Educ. (ICCSE)*, Aug. 2017, pp. 47–50.
[60] X. Zhao, X. Li, and J.-Y. Yin, "An empirical study on the mobile informatization teaching model applied to college students mental health education course," in *Proc. 1st Int. Cognit. Cities Conf. (IC3)*, Aug. 2018, pp. 295–298.
[61] J.-L. Liu and G.-D. Zhao, "The measurement and determinants of student satisfaction of education informatization in Chinese universities," in *Proc. Int. Symp. Educ. Technol. (ISET)*, Jul. 2018, pp. 183–187.
[62] H. Tsukamoto, Y. Takemura, H. Nagumo, I. Ikeda, A. Monden, and K.-I. Matsumoto, "Programming education for primary school children using a textual programming language," in *Proc. IEEE Frontiers Educ. Conf. (FIE)*, Oct. 2015, pp. 1–7.
[63] R. Schulz, G. M. Isabwe, and F. Reichert, "Investigating teachers motivation to use let tools in higher education," in *Proc. Internet Technol. Appl. (ITA)*, 2015, pp. 62–67.
[64] J. Schobel, R. Pryss, T. Probst, W. Schlee, M. Schickler, and M. Reichert, "Learnability of a configurator empowering end users to create mobile data collection instruments: Usability study," *JMIR mHealth uHealth*, vol. 6, no. 6, p. e148, Jun. 2018.
[65] B. Baruah, T. Ward, and N. Jackson, "Is reflective writing an effective peer assessment tool for students in higher education?" in *Proc. 16th Int. Conf. Inf. Technol. Based Higher Educ. Training (ITHET)*, Jul. 2017, pp. 1–6.
[66] U. Rosyidah, H. Haryanto, and A. Kardianawati, "Usability evaluation using GOMS model for education game play and learn English," in *Proc. Int. Seminar Appl. Technol. Inf. Commun. (iSemantic)*, Semarang, Indonesia, 2019, pp. 1–5.
[67] J. M. Arteaga and D. I. P. Rivera, "A process model to develop educational applications for children with dyslexia," in *Proc. 6th Int. Conf. Softw. Eng. Res. Innov. (CONISOFT)*, Oct. 2018, pp. 79–87.
[68] R. Delgado, S. Yacchirema, D. Quiroz, and Y. A. Liger, "Proposal to improve the usability of multiplatform mobile applications inside 'MiESPE' web portal at universidad de las fuerzas armadas," in *Proc. 13th Iberian Conf. Inf. Syst. Technol. (CISTI)*, Jun. 2018, pp. 1–4.
[69] R. M. Al-sumaty, "Data management system usability," in *Proc. Int. Conf. Smart Comput. Electron. Enterprise (ICSCEE)*, 2018, pp. 1–8.
[70] R. L. S. Putra, J. C. Patty, and D. B. Setyohadi, "Interaction design consideration for senior high school students: A usability evaluation of Go-Jek mobile applications," in *Proc. Int. Seminar Appl. Technol. Inf. Commun.*, Sep. 2018, pp. 196–201.

[71] S. C. Yang, T. L. Lee, and T. T. Feng, ''User experience of mobile application's interface: Measurement development,'' in *Proc. 5th Multidisciplinary Int. Social Netw. Conf.*, 2018, pp. 1–5.
[72] K. Alaa, H. Ali, and A. Wid, ''Predicting student performance in higher education institutions using decision tree analysis,'' *Int. J. Interact. Multimedia Artif. Intell.*, vol. 5, pp. 26–31, Jan. 2018.
[73] F. Meiland, A. Innes, G. Mountain, L. Robinson, H. van der Roest, J. A. García-Casal, D. Gove, J. R. Thyrian, S. Evans, R.-M. Dröes, F. Kelly, A. Kurz, D. Casey, D. Szcześniak, T. Dening, M. P. Craven, M. Span, H. Felzmann, M. Tsolaki, and M. Franco-Martin, ''Technologies to support community-dwelling persons with dementia: A position paper on issues regarding development, usability, effectiveness and cost-effectiveness, deployment, and ethics,'' *JMIR Rehabil. Assistive Technol.*, vol. 4, no. 1, p. e1, Jan. 2017.
[74] D. Roldán-Álvarez, E. Martín, M. García-Herranz, and P. A. Haya, ''Mind the gap: Impact on learnability of user interface design of authoring tools for teachers,'' *Int. J. Hum.-Comput. Stud.*, vol. 94, pp. 18–34, Oct. 2016.
[75] J. Zou, L. Xu, W. Guo, M. Yan, D. Yang, and X. Zhang, ''Which non-functional requirements do developers focus on? An empirical study on stack overflow using topic analysis,'' in *Proc. IEEE/ACM 12th Work. Conf. Mining Softw. Repositories*, May 2015, pp. 446–449.
[76] C. Y. Chong, P. Thongtanunam, and C. Tantithamthavorn, ''Assessing the students' understanding and their mistakes in code review checklists: An experience report of 1,791 code review checklist questions from 394 students,'' in *Proc. IEEE/ACM 43rd Int. Conf. Softw. Eng., Softw. Eng. Educ. Training (ICSE-SEET)*, May 2021, pp. 20–29.
[77] J. G. Meinema, N. Buwalda, F. S. van Etten-Jamaludin, M. R. M. Visser, and N. van Dijk, ''Intervention descriptions in medical education: What can be improved? A systematic review and checklist,'' *Academic Med.*, vol. 94, no. 2, pp. 281–290, Feb. 2019.
[78] R. D. Watts and I. W. Li, ''Use of checklists in reviews of health economic evaluations, 2010 to 2018,'' *Value Health*, vol. 22, no. 3, pp. 377–382, Mar. 2019.
[79] S. S. K. Chan, J. Geng, M. S.-Y. Jong, and D. T. M. Lau, ''Addressing the challenges in engineering classes: Harnessing active learning in a robotics course,'' in *Proc. Int. Symp. Educ. Technol. (ISET)*, Jul. 2018, pp. 162–164.
[80] S. B. Ayed, Z. Elouedi, and E. Lefevre, ''ECTD: Evidential clustering and case types detection for case base maintenance,'' in *Proc. IEEE/ACS 14th Int. Conf. Comput. Syst. Appl. (AICCSA)*, Hammamet, Tunisia, Oct. 2017, pp. 1462–1469.
[81] S. Bennani, A. Maalel, H. B. Ghezala, and M. Abed, ''Towards a decision support model for the resolution of episodic problems based on ontology and case bases reasoning: Application to terrorism attacks,'' in *Proc. IEEE/ACS 14th Int. Conf. Comput. Syst. Appl. (AICCSA)*, Hammamet, Tunisia, Oct. 2017, pp. 1502–1509.
[82] C. Bo, C. Gang, F. Tianxiang, and S. Feixin, ''Student affairs management system design based on mobile application EAider,'' in *Proc. 12th Int. Conf. Comput. Sci. Educ. (ICCSE)*, Houston, TX, USA, Aug. 2017, pp. 397–400.
[83] D. A. Prosvirin and V. P. Kharchenko, ''Model-based solution and software engineering environment for UAV critical onboard applications,'' in *Proc. IEEE Int. Conf. Actual Problems Unmanned Aerial Vehicles Develop. (APUAVD)*, Oct. 2015, pp. 312–315.
[84] J. Ke, O. Shopen, F. Li, W. Li, and M. Chang, ''Design based inspection methodology and application in the fab,'' in *Proc. China Semiconductor Technol. Int. Conf.*, Mar. 2015, pp. 1–3.
[85] F.-Z. Berriche, B. Zeddini, H. Kadima, and A. Riviere, ''Combining case-based reasoning and process mining to improve collaborative decision-making in products design,'' in *Proc. IEEE/ACS 12th Int. Conf. Comput. Syst. Appl. (AICCSA)*, Nov. 2015, pp. 1–7.
[86] Y. Lu and J. Saniie, ''Model-based parameter estimation for defect characterization in ultrasonic NDE applications,'' in *Proc. IEEE Int. Ultrason. Symp. (IUS)*, Oct. 2015, pp. 1–4.
[87] L. Gherardi and N. Hochgeschwender, ''Poster: Model-based run-time variability resolution for robotic applications,'' in *Proc. IEEE/ACM 37th IEEE Int. Conf. Softw. Eng.*, May 2015, pp. 829–830.
[88] D. Kirkpatrick, ''Techniques for evaluation training programs,'' *J. Amer. Soc. Training Directors*, vol. 13, pp. 21–26, Jan. 1959.
[89] L. Moldovan, ''Training outcome evaluation model,'' *Proc. Technol.*, vol. 22, pp. 1184–1190, Jan. 2016.
[90] *Malaysia Education Blueprint 2013–2025*, Ministry of Education Malaysia, Putrajaya, Malaysia, 2013.
[91] H. Baharudin, ''Pengajaran dan pembelajaran koperatif abad ke 21 Ismal-'adad wa al-ma' dud di Sekolah Menengah,'' Kolokium Pendidikan Bahasa Arab, Universiti Kebangsaan Malaysia, Malaysia, Tech. Rep., 2016.
[92] D. R. Anderson and L. W. Krathwol. (2001). *REVISED Bloom's Taxonomy Action Verbs: A Taxonomy for Learning, Teaching, and Assessing*. [Online]. Available: http://www.apu.edu/live./blooms_taxonomy_action_verbs.pd
[93] H. Noorbhai and T. Noakes, ''An evaluation of the coaching methods of the batting backlift technique in cricket,'' *J. Qualitative Res. Sports Stud.*, vol. 12, no. 1, pp. 35–56, 2018.
[94] N. Ngadiman, S. Sulaiman, N. Idris, H. Mohamed, A. R. A. Samad, and S. M. Osman, ''A survey on quality factors in designing educational applications for active learning,'' in *Proc. IEEE 9th Int. Conf. Syst. Eng. Technol. (ICSET)*, Oct. 2019, pp. 23–28.
[95] I. Stockwell, ''Introduction to correlation and regression analysis. statistic and data analysis,'' *SAS Global Forum*, vol. 364, pp. 1–8, Jan. 2008.
[96] J. Fox, *Regression Diagnostics: An Introduction*. Newbury Park, CA, USA: Sage, 2019.
[97] J. Hauke and T. Kossowski, ''Comparison of values of Pearson's and Spearman's correlation coefficients on the same sets of data,'' *Quaestiones Geographicae*, vol. 30, no. 2, pp. 87–93, 2011.
[98] M. Schrepp, ''On the usage of Cronbach's Alpha to measure reliability of UX scales,'' *J. Usability Stud.*, vol. 15, no. 4, pp. 1–12, 2020.
[99] J. Neyman and E. S. Pearson, ''On the use and interpretation of certain test criteria for purposes of statistical inference. Part I,'' *Biometrika*, vol. 20, no. 1, pp. 1–66, 2020.
[100] J. Neyman and E. S. Pearson, ''Contributions to the theory of testing statistical hypotheses,'' in *Joint Statistical Papers*. Berkeley, CA, USA: University of California Press, 2020, pp. 203–239.
[101] G. Kateman, ''Sampling theory,'' *Chemometrics Intell. Lab. Syst.*, vol. 4, no. 3, pp. 187–199, 1988.
[102] B. George, ''Sampling theory,'' *Int. Res. J. Modernization Eng. Technol. Sci.*, vol. 3, no. 6, pp. 1810–1815, 2021.
[103] K. Yamane, *Statistics: An Introductory Analysis*, 2nd ed. New York, NY, USA: Harper and Row, 1957.
[104] D. Koswara, S. Hardhienata, and R. Retnowati, ''Increasing teachers organizational commitment through strengthening teamwork, situational leadership and self-efficacy,'' *J. Ind. Eng. Manag. Res.*, vol. 2, no. 4, pp. 228–238, 2021.
[105] I. T. Godwill, O. B. Owei, and I. Brown, ''An analysis of existing housing stock in selected neighbourhoods in port Harcourt municipality,'' *Eur. J. Environ. Earth Sci.*, vol. 2, no. 3, pp. 57–61, Jun. 2021.
[106] A. Hernández, M. D. Hidalgo, R. K. Hambleton, and J. Gómez-Benito, ''International test commission guidelines for test adaptation: A criterion checklist,'' *Psicothema*, vol. 3, no. 3, pp. 390–398, 2020.
[107] S. Chacón-Moscoso, M. T. Anguera, S. Sanduvete-Chaves, J. L. Losada, and J. A. M. L.-L. Portell, ''Methodological quality checklist for studies based on observational methodology (MQCOM),'' *Psicothema*, vol. 31, no. 4, pp. 458–464, 2019.
[108] S. Aziz, M. Mahmood, and Z. Rehman, ''Implementation of CIPP model for quality evaluation at school level: A case study,'' *J. Educ. Educ. Develop.*, vol. 5, no. 1, p. 189, May 2018.

NORZILA NGADIMAN (Member, IEEE) received the bachelor's and M.Phil. degrees in software engineering and the Ph.D. degree in computer science from Universiti Teknologi Malaysia (UTM), in 2015, 2017, and 2022, respectively, under the supervision of Dr. Shahida Sulaiman. Her Ph.D. work was co-supervised by Dr. Norsham Idris and Dr. Hasnah Mohamed. She has received the South East Johor Development Authority (KEJORA) endowment that supports this research under the Cost Centre 4B349, Centre for Advancement in Rural Education Informatics (iCARE) UTM-KEJORA and partly from the contract research (4C555). The center provides the platform to conduct a community-based participatory research for the study. Her research interests include quality, usability, user interface aesthetic, learnability, understandability, appropriateness recognizability, satisfaction, effectiveness, web application, educational application, systematic literature review, and empirical software engineering.

SHAHIDA SULAIMAN (Member, IEEE) received the M.Sc. degree in computer science–real time software engineering and the Ph.D. degree in computer science from Universiti Teknologi Malaysia (UTM), Malaysia, in 2000 and 2004, respectively. She has been an Associate Professor with the Faculty of Computing, UTM, since 2011. Formerly, she had worked at Universiti Sains Malaysia (USM) for seven years. She has published and coauthored numerous technical papers mainly in software engineering area. She has been an editor for a number of journals, book chapters, and conference proceedings, besides serving as a reviewer and a technical committee member mainly in IEEE conferences, including Scopus and ISI indexed proceedings and journals in related fields. Her research interests include software design and architecture, software visualization and evolution, knowledge management, and education informatics.

NORSHAM IDRIS (Member, IEEE) received the M.Sc. and Ph.D. degrees in computer science from Universiti Teknologi Malaysia (UTM), Malaysia, in 2000 and 2015, respectively. She has been a Senior Lecturer with the Faculty of Computing, UTM, since 2016. Formerly, she had worked at UTM as a Tutor and a Lecturer for 19 years. She has published and coauthored numerous technical papers mainly in the artificial intelligence area. She has been a reviewer for a number of journals, book chapters, and conference proceedings. Her research interests include software engineering, intelligent and adaptive e-learning, and other artificial intelligence related research.

MOHD RAZAK SAMINGAN received the M.Sc. degree in computer science–distributed multimedia system from the University of Leeds, U.K., in 1998, and the Ph.D. degree in computer science from Universiti Teknologi Malaysia (UTM), Malaysia, in 2017. He is currently working as a Senior Lecturer with the Faculty of Computing, UTM. He is active in consultation for software development projects especially in web-based and mobile systems for several industry partners in Johor, Malaysia. His research interests include web technology, software architecture, and application framework.

HASNAH MOHAMED (Member, IEEE) received the M.Ed. degree in education technology and the Ph.D. degree in educational technology from Universiti Teknologi Malaysia (UTM), Malaysia, in 2002 and 2012, respectively. In 2013, she joined UTM as a Senior Lecturer. She has been working as an Associate Professor with the School of Education, Faculty of Social Science and Humanities, UTM, since 2021. Formerly, she had worked as a Secondary School Teacher for 16 years. She has published and coauthored numerous technical papers mainly in the education area. She has been an editor for several journals, book chapters, and conference proceedings, besides serving as a reviewer and a technical committee member of journals in the related fields. Her research interests include computer-assisted learning, computer-assisted instruction, flipped classroom, gamification, and other education related research.