

# Re-CRUD Code Automation Framework Evaluation using DESMET Feature Analysis

Asyraf Wahi Anuar<sup>1</sup>, Nazri Kama<sup>2</sup>, Azri Azmi<sup>3</sup>, Hazlifah Mohd Rusli<sup>4</sup>, Yazriwati Yahya<sup>5</sup>

Advanced Informatics Department, Razak Faculty of Technology and Informatics

Universiti Teknologi Malaysia, 54100 Kuala Lumpur, Malaysia<sup>1, 2, 3, 4, 5</sup>

Faculty of Information Management, Universiti Teknologi MARA Cawangan Johor, Kampus Segamat, Johor, Malaysia<sup>1</sup>

**Abstract**—A unified view of web application design and development is crucial for dealing with complexity. However, the literature proposes many denominations, depending on the development methodology, frameworks or tools. This multitude of Create, Read, Update and Delete (CRUD) approaches does not allow a holistic view of the web application. Besides, in a web application, the search for good practice in design, features and essential functions is still a relevant issue. A subset of essential CRUD operations is to provide code automation for web application rapid prototyping. Re-CRUD articulates the records management features into CRUD operation. This study aims to provide insight into the effectiveness and efficiency of Re-CRUD in web application development and to compare it with other web application frameworks' CRUD output. The qualitative feature analysis is used based on the evaluation guideline proposed in DESMET and reviewed by experts for validation. A document management system is developed and used as a case study for Re-CRUD evaluation. The feature analysis comprises Re-CRUD and four other web application frameworks CRUD, namely, CakePHP, Laravel, Symfony and FuelPHP. According to the review, Re-CRUD satisfies its expectations by providing more useful features and delivering higher code automation in the web application development process. Compared to the other existing CRUD generator, Re-CRUD has integrated records management features that are useful in providing support in managing born-digital data and also contributes to effectiveness and efficiency in web application development.

**Keywords**—Re-CRUD; web application; DESMET feature analysis; electronic records features

## I. INTRODUCTION

Web technologies have significantly influenced web application (WA) development and information system. The innovation of web technologies has allowed software developers and engineers to develop responsive cross-platform web applications rapidly. These technologies include creating read update delete (CRUD) generator, web application framework (WAF) and libraries that promotes reusable codes, rapid development and feasible features. WA can be considered a software component that stores and manages information just like a traditional information system but uses explicitly the web paradigm and associated technologies [1], [2]. It is a software system whose primary purpose is to publish and maintain data using hypertext-based principles [3], [4]. WA offers ease of access, maintenance, and cross-platform compatibility compared to the traditional desktop application installed on a local computer [5]. WA can be characterized as one that uses Web architecture and other technologies

(database, browser) to construct an information system that serves organizational needs [6].

The technique of WA development has rapidly changed over these past few years with the born of WAF, which promotes better development experiences and resources management [7], [8]. The WA development has become easier with the adoption of the frameworks, and lots of web-based innovation has been produced by non-technical people due to the framework innovation [9]–[13]. The WAF has bridged the possibility of WA development by a non-programmer and unlocks many potential new software possibilities and ideas [9], [14]. Most of the WAF embraces the MVC architecture that supports rapid and parallel development through CRUD operation, asynchronous technique, and straightforward business logic implementation, making the WA development more practical [9]–[12].

CRUD is the four fundamental components that manage the web application (WA) resources [12], [13], [15]–[18]. The create component allows the user or the WA itself (or both) to add a new data item to the database. Read component is used to retrieve the items recorded in the database and render them into a web page. The update component enables the user or the WA to edit an existing item and have those changes written back to the database. The last component, delete, enable the user or the WA to remove an item record from the database [19].

The CRUD paradigm is widely used among WA developers since it allows them to construct basic WA routine code and define how items in WA are related to one another [20]–[22]. CRUD is a provision of assistance in code generation and basic functionalities to support the developer in accomplishing the task [23]. CRUD enables the developer to create a quick-start application to work as the foundation of the WA solution [7], [24]. CRUD is an excellent technique to start an MVC-based WA project as it provides automation in design patterns [25]. Further, CRUD is a handy time-saver. It generates the skeleton codes for the WA and enables the developer to get faster output and demonstrate the WA prototype (input, process and output) to the WA project stakeholder [26].

The implementation of CRUD in WA development provides a substantial productivity boost for developers [12], [13], [15]–[17]. Using CRUD, the developer does not have to worry about many of the subtle details of wiring up the controller for the MVC application [12], [13], [27]. Although it boosts the development process, the traditional CRUD

generator only generates the fundamental functions that still present problems: their inadequacy to deal with the form features, authentication, search, file management, and others. [12], [13], [28]–[30]. A standalone CRUD could not satisfy the development of decent and complex WA since it lacks standard modern WA features to support the functions such as authentication, authorization, files management, search, internationalization, form features, report, logging and others [12], [13], [29]. Further enhancement and manual code modification are required to improvise the half-baked generated CRUD, especially in the integration of the time-consuming features and comprise repetitive coding for each of the CRUD output [12].

There is also an argument that the CRUD operation is not yet a complete solution for web application development. Many redundant tasks include repetitive code modification for feature integration after generating the CRUD [12]. Coding the same routine code for WA features repeatedly takes a long time and increases development costs [31]. However, it has a huge potential to go beyond the limit where it can automate more components for WA development [32]. The primary purpose of this study is to provide insight into Re-CRUD effectiveness and efficiency in web application development using the qualitative feature analysis case study based on the proposed guide in DESMET [33]. The contributions of this work are as follows:

- This work implements the records management features into CRUD operation for web application development.
- This study analyzes the effectiveness and efficiency of Re-CRUD in providing more features for web application development.
- This work compares the features and output of Re-CRUD and other aristocratic CRUD generators in web application frameworks.

The article is organized as follows. Section 2 describes the main related studies that have been conducted in the field of web application development challenges and Re-CRUD. Section 3 presents the methodology, including the DESMET feature analysis procedure and the instrumentation. Section 4 describes and analyzes the results. Section 5 presents a brief discussion. Finally, Section 6 presents the conclusions.

## II. LITERATURE REVIEW

### A. Issues and Challenges in Web Application Development

Web application (WA) development is a complex and challenging task since it requires consideration of numerous factors and requirements, some of which may contradict [34]–[42]. Many researchers have widely discussed the scalability issues, and developing a WA that scales well is a challenge [34], [43], [44]. As the WA is becoming mission-critical, there is a greater demand for it to improve reliability, effectiveness, performance, integration and security [34], [45]–[48]. To comply with the diverse expectations and requirements of many different users with varying skills and knowledge is very challenging. Most of the end-users are visual-oriented, which focuses on WA having more multimedia elements instead of

focusing on the functions of WA. They expect to manage and find the information they need faster using WA [6], [34].

Most WA is designed with a WAF with multi plugins and third-party scripting to enhance the features and a database as the storage medium [24], [26], [49]. Integrating plugins requires extensive knowledge of the amalgamation of the plugin and the WA due to the different programming languages and processes [50]–[52]. The integration is crucial as it will affect the WA's performance and stability, including the WA's functionality [34]. The developer must ensure the plugin's compatibility with the WAF and standard web browser technology [49], [53], [54]. Incompatible plugin integration may lead to vulnerabilities and security issues for WA due to deprecated methods or coding, leading to data corruption and unstable WA processing [55]–[58].

Due to the overwhelming number of WAFs published can be a daunting task to determine the most appropriate WAF [59]. The most important is that it can increase the programming productivity through code and files generation automation using CRUD operation, security advantages, and open-source that will impact the cost, support and documentation [60]. The WAF selection is vital as the speed and quality of work depend on it [60], [61].

The usability features issues in WA are design layout, design consistency, accessibility, information content, navigation, personalization, performance, reliability, filtering, analysis and design standards [62]–[66]. User response reveals that the search functionality, consistent navigation throughout the system, authentication, authorization, responsiveness and data visualization and reporting aid most WA usability [34], [67], [68]. Search features have been vital components in WA as the rapid trends in born-digital data and information lead to information overload and exposure [69]. The proper search function enables the user to filter the relevant information based on the search query [69]–[71]. The search function is a complex component where the developer needs to understand the filtering algorithm to ensure that the search results respond to the requested queries [11], [72]. The common issue in the search function is unable to satisfy the user query request due to the incomplete filtering algorithm [34], [62].

Localization is adapting WA (regularly written in English) for use in other countries, considering their culture, standard, regulations, principles and technology conditions [34], [73], [74]. Localization is more than just a language translation, and WA needs to be precisely designed to accomplish this multifaceted condition [73], [75], [76]. The scalability, reliability, availability, maintainability, upgradability, usability, speed and security are the terms used to describe how well the WA meets current and future needs. These ilties describe WA architectural qualities [34], [46], [77]–[79].

The design and development of WA for mobile compatible and device-independent operations are very complex and challenging. It must address various additional aspects compared to the traditional information system or desktop application and needs to satisfy many different stakeholders besides the diverse range of users. Poorly designed and develop WA has a high probability of low performance or failure [80].

### B. Re-CRUD

Re-CRUD was generally derived from the absence of electronic records management important features in the WAF CRUD. The conventional WAF CRUD operation is limited to only producing fundamental functions for WA, which requires further modification to include the other useful WA features to ensure the content can be managed systematically. Re-CRUD integrates the CakePHP framework CRUD operation with electronic records management important features that can produce additional useful functions and code automation. The following electronic records management important features are included in Re-CRUD [81]:

- Inventory: a descriptive listing of each record series or system, together with an indication of the location, access, and other pertinent data [82]–[84].
- Retention schedule: list how long each record series must be kept (the retention period), when the retention period starts (the cut-off), and the proper way to dispose of the record once retention is met (the disposition method) [84]–[87].
- Appraisal: the process of determining the archival value and ultimate disposition of records. Appraisal decisions are informed by several factors, including the historical, legal, operational, and financial value of the records [88]–[90].
- Disposition: the process of destruction of records or the transfer of records to another entity (most commonly an archival repository) for permanent preservation [91], [92].
- Role-based access control (RBAC): provides a role-based access control mechanism to offer protection from unauthorized access. Authenticated users with different roles have different authorization or access to the records [93]–[95].
- Search and retrieval: Enables the user to locate and retrieve records based on specific metadata, words or phrases. It is vital in any WA as it enables fast data retrieval via the search parameter [96], [97].
- Audit trail: provides log tracking for any changes to the electronic records to ensure validity and integrity [98], [99].
- Digital Archiving: transfer and store the valuable records into a repository that makes it non-active but still accessible through the system. It also helps to reduce the cluttered old and non-active records from the system [100], [101].
- Sharing: it provides the ability to transfer the record (internal to external or external to internal) in a single data or bulk data. There are several suggested formats such as CSV, XML and JSON [102], [103].
- Reporting: It summarises the current status of records such as total, active, inactive and the required appraisal attention and others [104], [105].

- Others: focused on the front-end framework for UI, data visualizer for reporting page, jQuery, DOMPDF and others supporting UI features. [82], [106].

Fig. 1 provides a simplified overview of Re-CRUD design based on the CRUD evolution using the console framework. It shows the evolution of the CRUD generation based on the traditional CRUD introduced in 1983 and is mainly used for the database abstraction process [17]. The technology continues to evolve into other application development segments, such as web application CRUD.

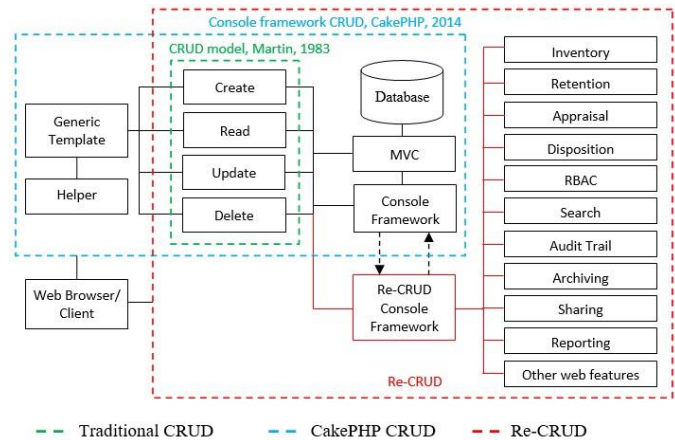


Fig. 1. Re-CRUD Console Framework.

## III. METHODOLOGY

### A. DESMET Feature Analysis

The DESMET method is designed to assist an evaluator in preparing and carrying out an unbiased and reliable assessment exercise within the framework of software engineering approaches and tools [33]. Feature Analysis is the term used by DESMET to describe a qualitative evaluation. Feature analysis focused on recognizing the requirements for a particular task and mapping it to the tool that was used to solve the task. Feature analysis is qualitative because it requires a subjective assessment of the relative importance of different features and how well they have been implemented. The qualitative case study is a feature-based evaluation performed by someone who has used the tool on an actual project. There are three main processes involved in carrying out a feature analysis, i) selection of feature; ii) feature scoring and ranking; iii) analysis and result interpretation, as explained in section V. For this research, the feature analysis is executed by the i) author and; ii) selected respondents, to compare and validate the feature analysis scoring from author bias. The following sub-section discusses the DESMET feature analysis procedure.

1) *Selection of features*: For features selection, the Re-CRUD features are grouped into four categories: CRUD operation, RBAC, electronic records management, and others, as shown in Table I. The sub-features are categorized based on the domain of the function.

2) *Feature scoring and ranking*: The Judgement Scale and Interpretation (JIS) is another definition that must be completed using DESMET. As described by Kitchenham [33], the JIS evaluation is according to the evaluator's perception.

As shown in Table II, if the feature is fully supported, the JIS is 1; if the feature is partly supported, JIS is 0.5; if the feature is not supported, JIS is 0 and -1 if the features make things worst [33]. The importance of a feature can be determined by deciding whether it is mandatory or merely desirable.

A method or tool that lacks a mandatory feature is unacceptable [33], [107], [108]. Non-mandatory features are considered desirable. This viewpoint on importance leads to two evaluation criteria: one that decides whether or not a feature is mandatory and the other that determines the degree to which a non-mandatory feature is desired. A ranking method is used to identify the electronic record important features are vital or only a desire for WA. Table III present the features set important weightage to identify the most imperative integrated features.

TABLE I. FEATURE SELECTION

Features		Sub-Features		Description
F1	CRUD operation	SF01	Code automation	The automation of code generation for persistent storage
F2	RBAC	SF01	RBAC	Provides a role-based access control mechanism to offer web content protection from unauthorized access.
F3	Electronic Record Mgt	SF01	Inventory	a descriptive listing of each record series, together with an indication of the location, access and other pertinent data
		SF02	Search	Ability to locate and retrieve data/record based on specific metadata, keyword or phrases.
		SF03	Audit Trail	provides log tracking for any changes to the electronic records to ensure that validity and integrity
		SF04	Transfer & Sharing	enable transfer or share the data (internal to external or external to internal) in a single or bulk data.
		SF05	Report	provides a summary related to the current status of records, active or inactive.
		SF06	Retention	list how long each record series must be kept
		SF07	Appraisal	the process of determining the archival value and ultimate disposition
		SF08	Archive	the records will be permanently stored, inactive but accessible for future references
		SF09	Disposition	the process of permanent destruction of records.
F4	Other web features	SF01	UI, visual design	the UI and other supporting components that supports the web application design and data presentation

TABLE II. JUDGMENT SCALE & INTERPRETATION (JI SCORE) [33]

Generic scale point	Definition of scale point	Score
Make things worse	Cause Confusion. The way the feature is implemented makes it difficult to use and/or encourages incorrect use.	-1
No support	Fails to recognize it. The feature is not supported nor referred to in the user manual.	0
Partly support	The feature is supported indirectly, for example, by using other tool features in non-standard combinations.	0.5
Full support	The feature appears explicitly in the feature list of the tools and user manual. All aspects of the feature are covered.	1

TABLE III. FEATURE SET IMPORTANT WEIGHTAGE [33]

Acronym	Level of importance	Weightage
M	Mandatory	4
HD	Highly desirable	3
D	Desirable	2
N	Nice to have	1

3) *The feature set and total score:* DESMET is an assessment technique that requires the assignment of scores to features and sub-features such as sub-feature importance levels, feature weights, and judgement scales. According to Marshall [108], the author first determined each score. At the initial stage, DESMET is required to determine the important level for each sub-feature. The identified features and sub-features set important weighting used in the feature analysis is shown in Table IV.

TABLE IV. FEATURES AND SUB-FEATURES SCORE USED IN THE ANALYSIS

ID	Feature Set	SF-ID	Subfeature: level of importance	Feature set Importance Weighting
F1	CRUD operation	F1-SF01	Mandatory	4
F2	RBAC	F2-SF01	Highly Desirable	3
F3	Electronic Record Management	F3-SF01	Desirable	2
		F3-SF02	Mandatory	4
		F3-SF03	Mandatory	4
		F3-SF04	Desirable	2
		F3-SF05	Highly Desirable	3
		F3-SF06	Highly Desirable	3
		F3-SF07	Mandatory	4
		F3-SF08	Mandatory	4
		F3-SF09	Mandatory	4
F4	Other web features	F4-SF01	Highly Desirable	3

The level of importance, together with the weightage for each of the sub-features, is identified. A feature will receive the highest possible score if all of the features in the set are completely present or supported. The level of importance for the features is determined by their implication and significance [108]. These weighted scores can be summed to obtain a percentage (%) score for each list of features. The following equation is used to compute the rating score for a feature set:

$$future\ set\ score\ \% = \frac{\sum\ of\ feature\ set\ score}{maximum\ score} \times 100 \quad (1)$$

In Table IV, the feature set is divided into four, F1 comprise one sub-features (F1-SF01), F2 comprises one sub-features (F2-SF01), F3 comprises nine sub-features (F3-SF01 to F3-SF09), and the F4 comprises one sub-feature (F4-SF01). The average (weighted) rate scores for each feature set are used to calculate a general percentage score for each model. A normalized score (percentage) is utilized because the feature set's sub-features vary.

For this calculation, the feature set weighting is used in Table V. The values here emphasize support for code automation via CRUD operation (F1) and for the electronic records management features (F3). Other weightings could be used, perhaps to emphasize usability, as tools to support the proposed solution to become more mature. The overall score for each CRUD generator can be determined using the following equation:

$$Overall\ score = \frac{\sum_{i=1}^4 (W_i TP_i)}{\sum_{i=1}^4 (W_i)} \quad (2)$$

$W_i$  is the weighting for the  $i^{th}$  feature set and  $TP_i$  is the percentage (%) score for the  $i^{th}$  feature set.

TABLE V. FEATURE SET WEIGHTING

Feature Set	Weight
F1	0.4
F2	0.2
F3	0.3
F4	0.1

**B. Instrumentation**

The primary purpose of the evaluation is to provide insight into Re-CRUD effectiveness and efficiency. A qualitative feature analysis case study is used based on the evaluation of guidelines proposed in DESMET [33] and reviewed by experts for validation.

1) *Case study:* Electronic Document Management System (EDMS) is used as a case study to assess and validate Re-CRUD. EDMS is a software system used to manage (organize and store) different kinds of data, information, and records. For this research, the EDMS is focused on managing born-digital data where it should be able to:

- Capture and validate the data input.
- Protect content using authentication and authorization.

- Practice proper electronic records management for the content.
- Multi-device and platform friendly.

Fig. 2 shows the application module that is available in the EDMS. To ensure the content is protected, the user must register and authenticate before accessing the content. It also includes the authorization procedure. Most of the records management aspect is put into practiced in the document repository.

As discussed in the following sub-section, seven steps are involved in designing and administering the case study [52].

a) *Identify case study context:* The case study objective is to evaluate the effectiveness and efficiency of the proposed solution. Table VI [55]–[58] shows that effectiveness and efficiency are highlighted as usability features. In the context of WA development, Re-CRUD is used to generate codes and files to form a WA with integrated records management features, as highlighted in Table IV.

b) *Select the host projects:* The EDMS case study applies the potential electronic record important aspects and is used to evaluate its effectiveness and efficiency [109].

c) *Identify the method of comparison:* As a comparative method, a cross-platform comparison is performed. Four additional CRUD generators are used for the cross-platform comparison, one using Re-CRUD and the others using the existing CRUD generator. The characteristics of all other development methodologies and procedures will be the same.

d) *Minimize the effect of confounding factors:* A similar host project characteristic and data storage are adopted to minimize the confounding effects. The selected WAF CRUD also have the same programming language, shares the same web server environment and uses almost similar development architecture. The selected respondents also must have experience, knowledge and understanding of software development and the use of WAF together with the CRUD operation [110]. This ensures that they are familiar with the CRUD operation and can focus on the evaluation instead of learning how to perform the CRUD operation and reduce the learning curve. Besides, specific and detailed instruction on configuration, database schema, development method, and CLI command is provided for the case study development.

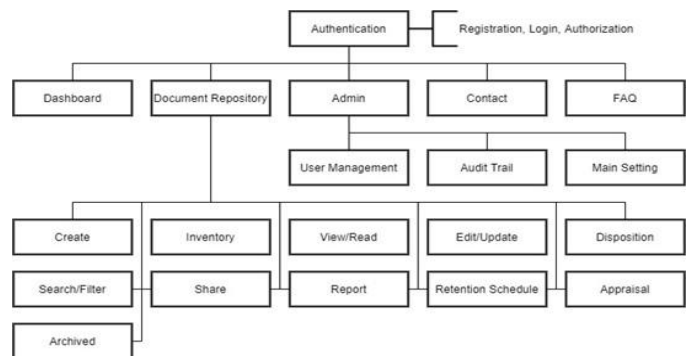


Fig. 2. EDMS Application Module used in Feature Analysis Case Study.

TABLE VI. SYSTEM USABILITY SCALE (SUS) QUESTIONNAIRE [115]

Usability aspect		Candidate item
Effectiveness (ES)	ES1	It allows me to accomplish my tasks. I think I would not need a system with more features for my tasks. I would not need to supplement Re-CRUD with an additional components. I found the system unnecessarily complex This system's capabilities would meet my requirements.
	ES2	
	ES3	
	ES4	
	ES5	
Efficiency (EC)	EC1	It saves me time when I use it. I found the various functions in this system were well integrated. I tend to reduce a lot of mistakes with this system. I don't make many errors with this system. I don't have to spend a lot of time correcting things with this system
	EC2	
	EC3	
	EC4	
	EC5	

e) *Plan the case study*: The following activities must be sequentially performed to complete the EDMS case study development. The listed activities require an understanding of software installation, configuration, MySQL database and PHP programming language.

- Gathering required files from the Github repository
- Host configuration
- Database migration and seeding
- Performed CRUD operation
- Output evaluation

f) *Executing the case study*: The author and respondents execute the development of the EDMS based on the given instruction and software development specifications. The respondents will fill in the online evaluation form at the end of the process.

g) *Analyze and report the results*: The score in the evaluation will be consolidated to identify the effectiveness and efficiency of the integrated electronic records management features in WA development.

2) *Expert validation*: Expert validation comprises opinion and judgement from the individual with knowledge and experience in the subject matter [111]–[113]. Expert validation is a methodology in which judgment is based on a particular set of requirements and/or experience obtained in a specific knowledge field, application area, product area, specific discipline, sector, and others [114]. The focus of this process is to validate the usability of the proposed solution. To execute the expert validation for this research, the experts must possess knowledge in software development, testing, maintenance and web-based technologies. The experts comprise representation from the industry, public sector and academician. The selected expert must respond to a set of questions as discussed in the following section.

3) *System usability scale questionnaire*: A set of questionnaires is provided to the respondents which are designed based on the System Usability Scale (SUS)

questionnaire [115]–[117]. The System Usability Scale (SUS) is an inexpensive yet effective and reliable tool for assessing the effectiveness and efficiency of a product [115]–[121].

The questionnaire is divided into two sections. First, the demographic comprises questions on the highest qualification, current working position, software development experiences, sector (mobile, web, IoT, desktop, cloud) and primary programming language. The second section will include the effectiveness and usability instrument and open-ended comments. The rating is based on 5 points Likert scale, which is anchored with one as strongly disagree and five as strongly agree [118], [122]–[124]. Table VI shows the usability aspect and the candidate item mentioned in the SUS questionnaire.

The SUS result is interpreted based on the grading scale, as shown in Fig. 3, to get a clearer picture of the effectiveness and efficiency of the proposed solution.

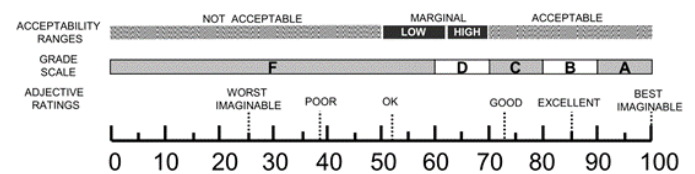


Fig. 3. The Adjective Ratings, Acceptability Scores and Grading Scales in Relation to the Average SUS Score [119].

The following equation is used to calculate the SUS score. The result will be interpreted based on Fig. 3.

$$sus\ score = \frac{(x \times 1) + (x \times 2) + (x \times 3) + (x \times 4) + (x \times 5)}{\text{total respondent} \times 5} \times 100 \quad (3)$$

#### IV. ANALYSIS AND RESULT

This section summarises the total scores for each CRUD generator for web application development. Table IV shows the feature weighting, and Table VII summarises the results of feature analysis for each of the WAF CRUD generators. This study aims to assess the effectiveness and efficiency of integrated records management aspects in CRUD operation from a web application development perspective. The feature analysis comprises five CRUD generators (embedded in WAF), i) Re-CRUD; ii) CakePHP; iii) Laravel; iv) Symfony and; v) FuelPHP. At the initial stage of DESMET evaluation, the importance level score has been given for each feature set as mentioned in Table IV and displayed in the sub-feature weightage score (C). The Judgment Scale and its Interpretation (JIS) is another definition that must be completed using DESMET. As described by Kitchenham [33], the JIS evaluation is according to the evaluator's perception.

Almost all sub-features in Re-CRUD have JIS = 1 since it is designed and developed based on the identified electronic records features important aspects, considering all intended features have been implemented. If the feature is fully supported, JIS is 1. If the feature is partly supported, JIS is 0.5. If the feature is not supported, JIS is 0 and -1 if the features make things worst [33]. Referring to Table VII, columns F, J, N, R and V show the JIS score for each sub-features of Re-CRUD, CakePHP, Laravel, Symfony and FuelPHP, respectively.

TABLE VII. FEATURE ANALYSIS RESULT

					Re-CRUD				CakePHP CRUD				Laravel CRUD				Symfony CRUD				FuelPHP CRUD														
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y											
Fea-Set	Sub-Fea	SF WS	MF SS	FI W	JI S	SF WS	F S S	FS S%	JI S	SF WS	F S S	FS S%	JI S	SF WS	F S S	FS S%	JI S	SF WS	F S S	FS S%	JI S	SF WS	F S S	FS S%											
F1	F1-SF-01	4	4	0.4	1	4	4	100	0.5	2	2	50	0.5	2	2	50	0.5	2	2	50	0.5	2	2	50											
F2	F2-SF-01	3	3	0.2	0.5	1.5	1.5	50	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0											
F3	F3-SF-01	2	30	0.3	1	2	28	93.33	1	2	9	30	1	2	8	26.67	1	2	9	30	0.5	1	5	16.67											
	F3-SF-02	4			1	4			0.5	2			0.5	2			0.5	2			0.5	2													
	F3-SF-03	4			0.5	2			0	0			0	0			0	0			0	0													
	F3-SF-04	2			1	2			0.5	1			0	0			0	0			0.5	1													
	F3-SF-05	3			1	3			0	0			9	30			0	0			8	26.67			0	0	9	30	0	0	0	0	5	16.67	
	F3-SF-06	3			1	3			0	0			0	0			0	0			0	0			0	0	0	0	0	0	0	0	0	0	0
	F3-SF-07	4			1	4			0	0			0	0			0	0			0	0			0	0	0	0	0	0	0	0	0	0	0
	F3-SF-08	4			1	4			0	0			0	0			0	0			0	0			0	0	0	0	0	0	0	0	0	0	0
	F3-SF-09	4			1	4			1	4			1	4			1	4			1	4			1	4	1	4	1	4	1	4	1	4	1
F4	F4-SF-01	3	3	0.1	1	3	3	100	0.5	1.5	1.5	50	0.5	1.5	1.5	50	0.5	1.5	1.5	50	0.5	1.5	1.5	50											
<b>Total Score</b>					<b>36.5/40</b>				<b>12.5/40</b>				<b>11.5/40</b>				<b>12.5/40</b>				<b>8.5/40</b>														
<b>Feature Set Weighting Overall Score (%)</b>					<b>88.00</b>				<b>34.00</b>				<b>33.00</b>				<b>34.00</b>				<b>30.00</b>														

SFWS - Sub-Feature Weightage Score (Mandatory - 4, Highly Desirable - 3, Desirable - 2, Nice to have - 1) [JIS\*SFWS(C)]  
 MFSS - Max-Feature Set Score [sum SFWS(C)]  
 FIW - Feature Important Weightage (0.4, 0.3, 0.2, 0.1)  
 JIS - Judgement Scale and Interpretation (Full supported - 1, Partly supported - 0.5, No support - 0, Make thing worst - -1)  
 FSS - Feature Set Score (sum of each SFWS)  
 FSS% - Feature Set Score Percentage (FSS/MFSS\*100)  
 Feature Set Weighting Overall Score - (F1 FSS%\*F1 FIW)+(F2 FSS%\*F2 FIW)+(F3 FSS%\*F3 FIW)+(F4 FSS%\*F4 FIW)

Based on the JIS score, the Sub-Features Weightage Score (SFWS) for each WAF CRUD is calculated (Re-CRUD (G); CakePHP (K); Laravel (O); Symfony (S) and; FuelPHP (W)).

The weight score is the multiplication of the JI with the respective Sub-Feature Weightage Score (SFWS) (C). For example, for F1-SF01, the SFWS for Re-CRUD is 4 (G), obtained by JIS(F)\*SFWS(C). Next, from the SFWS, the

Feature Set Score (FSS) is calculated (Re-CRUD (H); CakePHP (L); Laravel (P); Symfony (T) and; FuelPHP (X)). This FSS is the sum of the SFWS grouped by the feature set. For example, in column H, the FSS for Re-CRUD feature set 3 (F3) is 28 is the sum of (2+4+2+2+3+3+4+4+4). In the sequence, the FSS percentage (Re-CRUD (I); CakePHP (M); Laravel (Q); Symfony (U) and; FuelPHP (Y)) is obtained by dividing the respective FSS with MFSS (D) for each feature set.

Finally, the overall score for each WAF CRUD is calculated with the sum of each FSS percentage, considering the feature set and sub-feature set important levels (Table VII). For instance, considering the FIW for F1 is 0.4, and the FSS% for F1 Re-CRUD is 100% (I), the % of the feature set weighting score is  $100\% * 0.4 = 40\%$ . For F2, the weighting score is 0.2; then, it is calculated using  $50\% * 0.2 = 10\%$ , and so on. In the end, all these values are summed, and the feature set weighting overall score is obtained. Each feature set score will be described in-depth in the following subsection.

Based on the feature set weighting overall score in Table VII, Re-CRUD leads the scores with a massive margin of difference, Re-CRUD: 88%; CakePHP: 34%; Laravel: 33%; Symfony: 34% and; FuelPHP: 30%. Technically, this is because Re-CRUD is specifically designed and developed based on the identified electronic records features mentioned earlier. Rather than integrating the electronic records features into the CRUD operation, Re-CRUD also has reconstructed the existing CRUD operation functions and features to make it more systematic and ensure that each of the features and functions still exists in the code automation generation. The score is within 30% to 34% for the other CRUD generators because the electronic records features are not present in the CRUD operation. Even though some of the features are present, the JIS score is 0.5 (partly supported), requiring modification or enhancement to the generated files and coding. Hence, even though Re-CRUD is new compared to the other listed aristocratic WAF CRUD elite, the integrated electronic records features make it more effective and efficient in WA development, especially in managing the born-digital content in the WA. The integrated important aspects have crucial roles in managing digital content by promoting appropriate electronic records management functions. It enables the semi-active record to be appraised, archived the inactive record, disposed of unused records, reporting, and other features contributing to WA usability.

#### A. Feature Set 1: CRUD

The CRUD is focused on code automation, where it generates the fundamental function of WA based on the WAF architecture. The CRUD operation is embedded into the respective WAF as a plugin to enable the developer to generate the WA prototype rapidly. During the case study, it was found that all of the listed CRUD generators can produce WA fundamental components as expected. Referring to Fig. 4, the Re-CRUD FSS is 100%, and the other CRUD generators score 50%. Based on the evaluation, all of the CRUD generators can produce a skeleton of WA with fundamental functions. The JIS score for Re-CRUD is 1 (fully supported) compared to the other's score is 0.5 (partly supported). The Re-CRUD operation provides a more comprehensive solution in generating the files

and codes for WA where the additional features from electronic records features are fully integrated and generated using the same command without any additional modification made by the developer. The other WAF CRUD operation is limited to producing the fundamental components only and requires further modification. This modification is considered a manual code modification where the developer must reconstruct some of the CRUD generated code to enhance and integrate with other features or other third-party plugins.

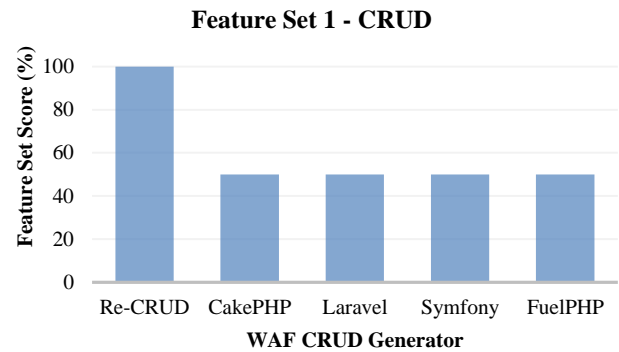


Fig. 4. Feature Set 1 - CRUD Operation (Code Automation) FSS.

From the developer's perspective, the electronic records feature integration is beneficial due to no code modification, reducing the repeating coding process, e.g., integrating search into different tables in WA and reducing the development timeframe. In accordance with the rapid application development methodology in CRUD operation and modern WA features requirements, the integrated electronic records features in Re-CRUD offer an effective solution for rapid development, fully-featured functions and code automation. The additional features generated in CRUD operation allow the developer to focus on the other vital features that the stakeholder requires.

Overall, based on the case study evaluation, all CRUD operations are working as expected and fully supported (code automation) to generate the fundamental component to form a WA. However, Re-CRUD can produce more functions to support the modern WA without having any additional extension or plugin. The MFSS for CRUD is 4 with FIW is 0.4, which considered that the CRUD operation is vital since it is the generator for code automation and crucial to providing a rapid WA development process.

#### B. Feature Set 2: Role-based Access Control (RBAC)

The RBAC is the authentication and authorization which enable the protection of WA data and content. The features are evaluated to determine whether the respective WAF CRUD can provide the functions without having manual code modification or a third-party plugin. The RBAC is categorized as highly desirable since it is considered one of the most important features that need to be available in WA. Fig. 5 shows the RBAC FSS where the Re-CRUD score is 50%, and the other WAF CRUD is 0%. The RBAC does not fully support the other WAF CRUD generator aptitude. Re-CRUD has embedded the RBAC into the CRUD operation where the process of the authentication data table is migrated and the seed



inside the same generator. However, the integrated RBAC in Re-CRUD is considered partly supported due to the RBAC environment is not entirely present where the authorization policy (who is allowed to access what) is not integrated with CRUD operation. The authorization policy is important to manage and process the user's permission to access a specific resource or function.

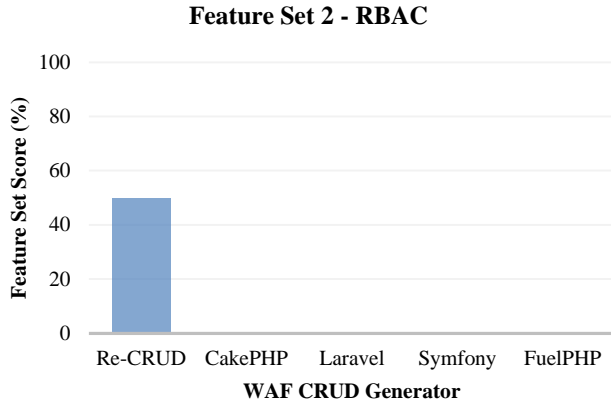


Fig. 5. Feature Set 2 – Role-based Access Control FSS.

The policy needs to be manually generated using the authorization policy. By default, the policy is available, but the CRUD operation did not update it. The developer is expected to have an option to generate the policy content during the CRUD process. This policy automation can be considered in the Re-CRUD future update.

### C. Feature 3: Electronic Records Management

Feature set 3 (F3) comprises 9 sub-features, which is, inventory (F3-SF-01); search (F3-SF-02); audit trail (F3-SF-03); transfer and sharing (F3-SF-04); reporting (F3-SF-05); retention (F3-SF-06); appraisal (F3-SF-07); archive (F3-SF-08) and; disposition (F3-SF-09). Fig. 6 shows the highest FSS is by Re-CRUD with a score of 93.33% (Table VII).

Re-CRUD (F3) FSS is 26.5/30, showing that Re-CRUD supports most electronic record features. Based on the JIS, it shows that all of the Re-CRUD F3 sub-features scores are fully supported (JIS = 1) except for F3-SF-03 (audit trail). It is due to the audit trail requiring more in-depth functions to support the audit log process, which enables the edited records to be reverted to the original state and it is considered partly supported (JIS = 0.5). For the CakePHP and Symfony CRUD operation, they share the same FSS score, 30 %, which is 3.33% higher than Laravel's 26.67% score. Compared to the Laravel FSS, the difference between CakePHP and Symfony is the transfer and sharing features where this SFWS is tagged as desirable (2). Both of the WAF CRUD operations partly support (0.5) the feature. FuelPHP FSS for F3 is 16.67%, where most electronic records features are not present. The inventory JIS for FuelPHP CRUD is partly supported (0.5), and for disposition, the JIS is fully supported (1), which brings the SFWS to 5/40. Fig. 7 shows the complete analysis for features 3 (electronic records) components.

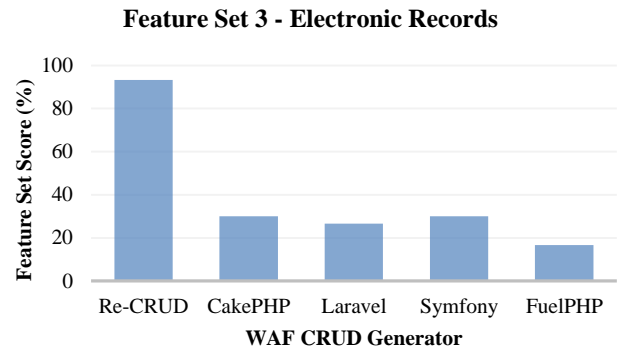


Fig. 6. Feature Set 3 – Electronic Records FSS.

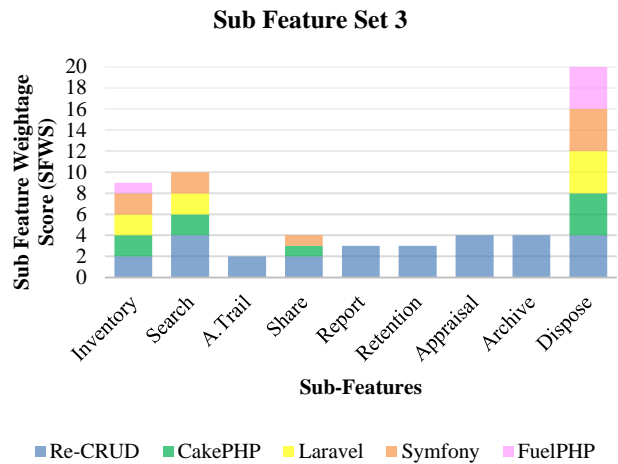


Fig. 7. Feature 3 SFWS (Electronic Records).

TABLE VIII. RESPONDENTS DEMOGRAPHIC DATA

Respondent	Age	Gender	Highest Academic Qualification	Software Development Experience	Software Application Major Sector	Primary Programming Language
R1	36-40	Female	Degree	> 6 years	Web	Java
R2	26-30	Male	Degree	3 - 6 years	Web	PHP
R3	36-40	Female	Master	> 6 years	Web	PHP
R4	36-40	Male	Degree	> 6 years	Web	Python
R5	41-45	Male	Degree	> 6 years	Web	Java
R6	31-35	Male	Degree	> 6 years	Cloud	PHP
R7	36-40	Male	Degree	> 6 years	Cloud	Java
R8	31-35	Male	Master	3 - 6 years	Web	PHP

The audit trails should have a comprehensive function in managing the data history or content changes. The Re-CRUD audit trail aims to preserve the content by tracking the series of changes in the data or records. By tracking the changes, the authenticity of the data or content can be preserved and enable the system administrator to identify the authorized person that amended the content.

The inventory (F2-SF-01) shows that most of the CRUD generator is fully supported (JIS = 1). Most of the CRUD generator produces the index file where this index lists all data or records in the web application, which serve the same purpose of inventory but without the record series. The search (F3-SF-02) shows that Re-CRUD JIS = 1, CakePHP, Laravel and Symfony JIS = 0.5 and FuelPHP JIS = 0. The search feature should be generated in the CRUD operation as the search feature is vital in providing fast access and retrieval. With the integration, Re-CRUD can generate the search function embedded in the inventory and all necessary functions. This is very useful for the developer where they can save more time and reduce the process of coding the search routine code in every single table in the WA. Search is not integrated with the operation for the other web CRUD generator but can be integrated using a plugin. However, the integration required more time and testing the integration.

The sharing feature (F3-SF-04) shows that Re-CRUD has fully supported (JIS = 1), CakePHP and Symfony are partly supported (JIS = 0.5), where the Laravel and FuelPHP JIS = 0 due to not being supported by regular CRUD operation. The sharing feature enables the user to share the content using a link, email or QR code. Re-CRUD generates the sharing function and embeds it into the inventory and view layer. The link and QR code is dynamically generated for each of the content in the web application. However, the access is still subjected to the RBAC to ensure that the content is protected and secure. Without the integrated sharing feature, the developer needs to code the functions for each of the tables, and it requires more time for the development. The reporting feature (F3-SF-05) is considered one of the most crucial features in a web application where it provides a summary of data and records. This feature is useful for the system administrator to populate the data in the web application and retrieve the report, for example, the monthly statistic of the data entry, active and inactive records, archive records and others. Re-CRUD has fully supported this feature (JIS = 1), where the reporting functions are integrated into each generated table. The integration also utilized the data visualization feature from F4-SF-01, where the ChartJS is used to generate the chart to make the report more systematic and readable. However, the reporting features are not supported by the other CRUD generator.

For the retention (F3-SF-06), appraisal (F3-SF-07) and archived (F3-SF-08), it was found that only Re-CRUD is fully supported (JIS = 1), and the other generator is not supported. Retention enables systematic record management in the web application. It provides a duration for each of the records before deciding to be disposed of or archived permanently. The retention feature works together with the appraisal feature (F3-SF-07). The appraisal feature provides the function to evaluate the specific record that has past the retention due date before deciding to be disposed of or archived. Each of these functions is generated through the CRUD operation using Re-CRUD. The digital archival feature in Re-CRUD enables the system administrator to move the inactive record that still has significant value, e.g. fiscal, legal, historical and other vital records, to permanent storage. The records that have moved to the archive are permanently stored, and the edit feature is

disabled to protect the originality and authenticity of the records. This feature is critical to ensure that the web application is not burdened by the unnecessary records, which may lead to an information explosion due to unmanageable data and records in the web application.

The disposition feature (F3-SF-09) shows that all CRUD generators have the feature. The objective of this feature is to remove or dispose of the records from the web application database. Technically, this feature is considered a standard feature in the CRUD operation. Based on the objective, all CRUD operations are fully supported (JIS = 1) since they can delete the records from the web application database.

#### *D. Feature Set 4: Other*

Feature set 4 focused on the UI and other supporting features to support the WA design and data presentation. Fig. 8 shows the FSS for each of the WAF CRUD. The weightage score for F4-SF01 is highly desirable since it is important to render the responsive UI, generate charts for data reporting, generate PDF, and provide a WYSIWYG editor. Technically, Re-CRUD is fully supported (100%) with those features and successfully integrated into the CRUD output. For the other WAF CRUD generator, the FSS is 50% since some of the features are available. Although it is not fully similar to the specification, it is still natively able to provide equivalent functions; for example, the template can still render responsively. PDF can be printed using the default print method and others.

#### *E. Expert Validation*

To validate the result from Table VII, feature analysis has been carried out with industry professionals' participation to evaluate Re-CRUD. The result is compared with the author's feature analysis result. Expert validation comprises opinion and judgement from the individual with knowledge and experience in the subject matter [111]–[113]. Expert validation is a methodology in which judgment is based on a particular set of requirements and/or experience obtained in a specific knowledge field, application area, product area, specific discipline, sector, and others [114]. The focus of this process is to validate the usability of the proposed solution. To execute the expert validation for this research, the experts must possess knowledge in software development, testing, maintenance and web-based technologies. The experts comprise representation from the industry, public sector and academician. The selected expert must respond to a set of questions as discussed in the following section.

To get a concrete outcome for the case study, it is important to know that the selected respondent must have knowledge in the specific area of the testing and understand how the product can provide a decent solution for their problem [125], [126]. Previous studies have suggested that five respondents are sufficient for usability testing, revealing 80% of usability issues [127]. Another usability researcher stated that a group size of 10 respondents is sufficient where it reveals a minimum of 82% of the problem [128]. A group size of 7 respondents is optimal for studies; even where the study is quite complex will reveal 95% of the problem [129]. It is also suggested that the respondents have expertise in a specific selected field, and the

optimal sample size also should be influenced by the study's complexity and diversity of the respondents [126], [128].

Based on the optimal number of respondents as suggested by previous studies, the usability test for this study is performed by seven selected respondents. The selected respondent is a practitioner that is active and knowledgeable in WA development, where they must-have experience in WA development using WAF and CRUD operation. Considering their experiences and knowledge, the respondents are experts in WA development and familiar with the task domain, WAF, MVC architecture, CRUD and other related features and functions.

The expert validation is executed to compare and validate the feature analysis scoring from the author's bias. 10 Re-CRUD feature analysis case study invitation has been emailed to selected experts, and eight have completed the case study and evaluation. The same case study has been executed by eight selected experts using the same procedure and analyzed. Table VIII shows the respondent demographic data. The demographic information shows that six respondents are male and two female. Their age is in the range of 26 to 45 years old. Six respondents have a bachelor's degree, and the other two respondents have a master's degree. In terms of software development experience, six of the respondents have more than six years of experience, while the others have 3 to 6 years of software development experience, and their software development major sector is in web and cloud. Four of the respondents used PHP as their primary programming language, three respondents used Java, and the others used Python.

Table VII shows the Re-CRUD feature set overall weighting score is 88%, based on the author's evaluation. Fig. 9 compares the Re-CRUD feature set weighting score given by the author and another eight selected experts. A based-line score (88%) has been set and marked with a dashed line compared to the other experts' scores.

Fig. 10 shows the feature analysis responses from industry professionals. Based on the feature set weighting overall score, it was found that 7 (87.5%) of the responses from the experts' score were higher compared to the based-line score (88%) and only 1 (12.5%) scored 0.5 below the based line. Most of the response is above the baseline as the listed features can be generated during the CRUD operation and performed as expected.

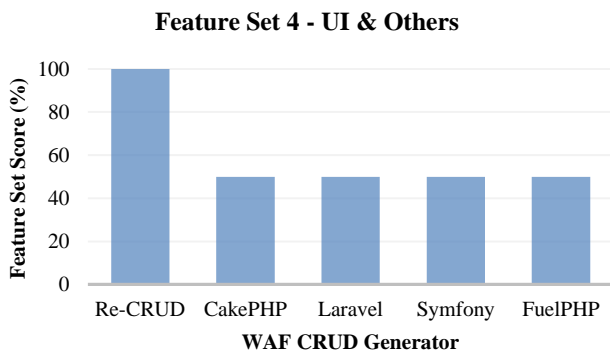


Fig. 8. Feature Set 4 – UI & Others FSS.

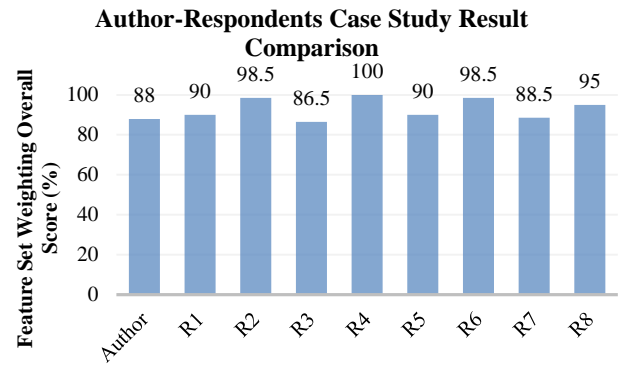


Fig. 9. Comparison of Re-CRUD Feature Set Weighting Score between Author and Respondents.

FS	SF	R1					R2					R3					R4				
		JIS	SFWS	FSS	FSS%	FIW%	JIS	SFWS	FSS	FSS%	FIW%	JIS	SFWS	FSS	FSS%	FIW%	JIS	SFWS	FSS	FSS%	FIW%
F1	F1-SF-01	1	4	4	100	40	1	4	4	100	40	1	4	4	100	40	1	4	4	100	40
F2	F2-SF-01	0.5	1.5	1.5	50	10	1	3	3	100	20	0.5	1.5	1.5	50	10	1	3	3	100	20
F3	F3-SF-01	1	2				1	2				1	2				1	2			
	F3-SF-02	1	4				1	4				1	4				1	4			
	F3-SF-03	1	4				1	4				0.5	2				1	4			
	F3-SF-04	1	2				1	2				1	2				1	2			
	F3-SF-05	1	3	30	100	30	1	3	28.5	95	28.5	0.5	1.5	26.5	88.33	26.5	1	3	30	100	30
	F3-SF-06	1	3				0.5	1.5				0.5	1.5				1	3			
	F3-SF-07	1	4				1	4				1	4				1	4			
	F3-SF-08	1	4				1	4				1	4				1	4			
	F3-SF-09	1	4				1	4				1	4				1	4			
F4	F4-SF-01	1	3	3	100	10	1	3	3	100	10	1	3	3	100	10	1	3	3	100	10
FSW Overall Score (%)		90					98.5					86.5					100				

Fig. 10. Feature Analysis Responses.

As highlighted in Fig. 10, several features in the feature analysis score are partly supported. For the F2-SF-01 (RBAC), four respondents (R1, R3, R5 and R7) stated the score was partly supported and tallied with the author has given scores. It was found that the F2-SF-01 (RBAC) feature is considered incomplete because some of the sub-features are not present. As stated in B, the RBAC is partly supported because the authorization policy is not integrated and generated with the CRUD operation. The RBAC policy enables the developer to manage who can access/restrict specific resources. However, four respondents are satisfied with the RBAC, which is fully supported since the authentication is fully functional and the authorization policy is not a compulsory component. Technically, the authorization policy can vary depending on the WA requirement. The authorization can be achieved using the authentication group and session through a simple programming procedure.

One response (R3) stated that F3-SF03 (audit trail) is partly supported. This is due to the issues of the ability to restore the original content from the audit trail history. Re-CRUD is built to have audit trail features that can provide digital tracking of content changes and present a list of changes history. With the

more complex concept of audit trail, R3 expect to have a restoration procedure that can revert specific changes to the original state. This feature is currently not present in Re-CRUD since it was designed to capture and provide the history of the changes. Although the restoration ability can be integrated, it requires a more complex audit trail management. For example, the restoration also needs to be tracked where the RBAC policy must be appropriately configured to ensure that only authorized persons can access the features to revert the content to the original state.

For the F3-SF05 (Report), one (R3) of the respondents stated this feature is partly supported due to the absence of a dynamic report generator. Technically, Re-CRUD provides a report based on the current year and monthly data. However, the report is currently unable to generate customized request reports, for example reports based on a specific date range. Furthermore, three (R2, R6 and R7) respondents stated that F3-SF06 (Retention) is partly supported. Re-CRUD retention features are designed with a specific duration. Even though it has various options of retention duration (6 months, 1, 3, 5 and 7 years), these options may not match the developer requirement due to different policies by the WA stakeholder. The retention duration options are currently not flexible and require manual code modification, and may cause a repetitive task.

As mentioned earlier, most of the feature set weighting overall score is higher than the baseline score. However, one of the respondents (R3) score is 86.5% which is 1.5% lower than the baseline score since R3 is not satisfied with the F2-SF01 (RBAC), F3-SF03 (audit trail), F3-SF05 (retention) where for the F3-SF03 (audit trail) features, the SFWS is 4 (mandatory) which affect the scores. As explained in the previous paragraph, each integrated feature has a scope of functions and limitations. It may not generally be complete; however, it can still provide basic functions for each feature.

#### F. System Usability Evaluation Result

Eight experts have executed system usability evaluation after completing the case study for the feature analysis. This evaluation aims to capture feedback on Re-CRUD effectiveness and efficiency in web application development. Ten system usability scale (SUS) [115] questions have been asked, which focused on two aspects (effectiveness and efficiency). The index for every question was calculated from equation 3, and the results are presented in percentages. Table IX shows the collective score for effectiveness and efficiency together with the SUS score percentage.

Based on Table IX, in the effectiveness aspect (ES1, ES2, ES3, ES4, ES5), respondents strongly agree that Re-CRUD is effective in completing their task in web application development (ES1) and agreeing that it meets their requirements and is easy to use (ES4, ES5). The score for ES2 is 85%, where the developer required additional features to complete their task due to the other special requirement for a specific project. However, Re-CRUD is designed to be flexible to be applied in various types of web application projects. In the aspect of supplementing Re-CRUD with additional components (ES3), the score is 82.5%, where the respondents feel that they are still required to supplement Re-CRUD with

other components, but most of the required components are already exist in the Re-CRUD environment. The additional suggested component is automation in data/content backup, error logging, and testing from the respondent comments. Fig. 11 shows the score for Re-CRUD effectiveness. Overall, Re-CRUD effectively performs code automation for web application development.

TABLE IX. RE-CRUD EFFECTIVENESS AND EFFICIENCY SCORE

			Score					SUS Score (%)
			1	2	3	4	5	
Effectiveness (ES)	ES1	It allows me to accomplish my tasks.	0	0	0	0	8	100
	ES2	I think I would not need a system with more features for my tasks.	0	0	0	6	2	85
	ES3	I would not need to supplement Re-CRUD with an additional components.	0	0	1	5	2	82.5
	ES4	I found the system unnecessarily complex	0	0	0	1	7	97.5
	ES5	This system's capabilities would meet my requirements.	0	0	1	1	6	92.5
Efficiency (EC)	EC1	It saves me time when I use it.	0	0	0	0	8	100
	EC2	I found the various functions in this system were well integrated.	0	0	0	1	7	97.5
	EC3	I tend to reduce a lot of mistakes with this system.	0	0	0	0	8	100
	EC4	I don't make many errors with this system.	0	0	0	1	7	97.5
	EC5	I don't have to spend a lot of time correcting things with this system	0	0	0	0	8	100

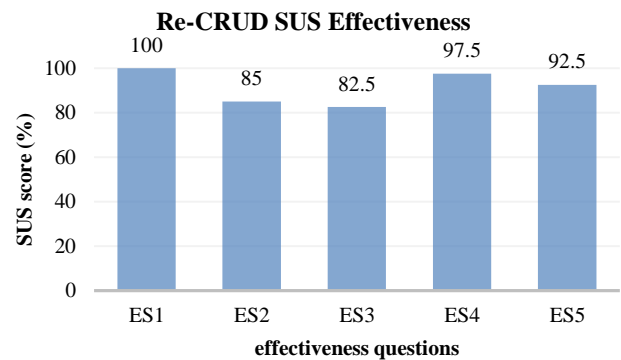


Fig. 11. Re-CRUD Effectiveness Score.

For the efficiency aspects (EC1, EC2, EC3, EC4, EC5), the respondents strongly agree with all questions in efficiency aspects. It can be seen from questions EC1 until EC5 that it has an indexed percentage above 95%. In terms of time consumption in completing respondent tasks (EC1, EC5), Re-CRUD can support rapid development, reducing the development time by providing more automation and reducing

the tendency to reduce mistakes in code writing due to human error. Re-CRUD also presented a high percentage of score in functions integration (EC2) and lower error from the default generated code automation (EC4). Fig. 12 shows the score for Re-CRUD efficiency, and it can be determined that Re-CRUD are efficient in reducing development time, systematic in features integration and reducing code error.

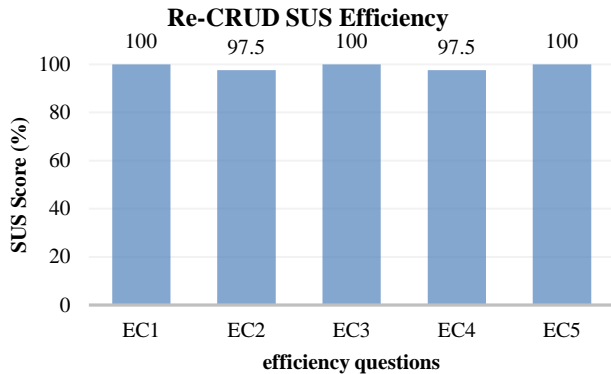


Fig. 12. Re-CRUD Efficiency Score.

## V. DISCUSSION

Three types of evaluation have been conducted (case study, feature analysis and expert validation), and the results are used as the basis of these findings. The case study is used to execute EDMS development using Re-CRUD to demonstrate the code automation with the integrated important aspects. From the web application development perspective, Re-CRUD can establish the EDMS with the integrated features. The code automation from Re-CRUD can be executed appropriately in the web server environment and produces no error when performing a task. Therefore, based on the analysis of the evaluation results, Re-CRUD has reached the following conclusions without hesitation:

### A. Integrated Web Application Important Features with Code Automation

The Re-CRUD output offers extra features and solutions for modern web applications. Technically the extra features can be considered must-have features since the most web application is designed to create, manage and maintain digital data or content. The records management features have been integrated into the Re-CRUD tested and compared with four other WAF CRUD generators. The feature analysis result shows that Re-CRUD has better CRUD operation output and offers more features than the other WAF CRUD generator.

### B. Reduce Web Application Development Time and Code Error

With the integrated features, Re-CRUD enables the developer to speed up the development process; since all of the modern WA necessary routine features have been generated, the developer can focus on other important functions that the stakeholder requires. Using Re-CRUD also enables the developer to build the prototype and test the WA rapidly. Expert validation shows that EC1, EC3, and EC5 have the most excellent efficiency scores (100%), where Re-CRUD may

save developers time and minimize the time in code debugging due to code automation. With more code and features automation, the tendency of prone to coding error due to human mistakes also decreases.

### C. Achieved Usability Attributes

Effectiveness and efficiency are the usability attributes identified for this research evaluation. An expert validation was commenced to assess the Re-CRUD with this attribute. Eight industry experts have been invited to participate in the validation process. The SUS questionnaire with 5 points Likert scale is used, and both of the usability attributes, effectiveness and efficiency result achieved the SUS acceptable score. Considering this result, there is no doubt that Re-CRUD is effectively and efficiently acceptable in handling WA development.

## VI. CONCLUSION

The overall score indicates that Re-CRUD with the integrated electronic records management features is effective and efficient in developing a WA. Overall, Re-CRUD receives a perfect score for feature sets 1 and 4, while the other CRUD generators receive a 50% score for the same feature set. Re-CRUD provides integrated RBAC for feature set 2, although it is only deemed partially supported because the other CRUD generator does not provide integrated RBAC. The electronic records management capabilities emphasize Feature Set 3, and the majority of the features have been effectively integrated into Re-CRUD, with a score of 93.3%. Although other CRUD generators cover some electronic records management features, the function is only partially supported. More code modification and enhancement are required for the other CRUD outputs to incorporate key electronic records management features. Re-CRUD allows the WA developer to create a rapid WA prototype with a more practical solution to digital content management through the integration of records management tools. The integrated records management features in CRUD operation enable the developer to save more time in coding the routine code for the web application to provide the essential features such as search and reporting in the web application. It makes the development process faster with the code automation for all the routine codes. This research also has a limitation where Re-CRUD only includes eleven features with a specific task to be evaluated. Many other features can be included, but only the listed features are integrated and tested to maintain flexibility. Re-CRUD has been applied and tested on EDMS, small size of WA. A larger scale of WA may behave differently as the data table will be more prominent, and the data processing may be more complex. In future work, more features should be incorporated into Re-CRUD, i.e., progressive web apps (PWA) and improve the Re-CRUD practical ability in various web application development. At the same time, it can effectively provide mobile application ability to the generated CRUD.

## ACKNOWLEDGMENT

The study was funded by the Encouragement Research Grant (Vote No. Q.K130000.3856.20J92) awarded by Universiti Teknologi Malaysia.

REFERENCES

- [1] G. J. Houben, P. Barna, F. Frascar, and R. Vdovjak, "Hera: Development of semantic web information systems," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 2722, pp. 529–538, 2003.
- [2] T. Isakowitz, M. P. Bieber, and F. Vitali, "Web Information Systems," *Commun. ACM*, vol. 41, no. 7, pp. 78–80, Jul. 1998.
- [3] A. Athanasiadis and Z. Andreopoulou, "A Web Information System Application on Forest Legislation: The Case of Greek Forest Principles," *Procedia Technol.*, vol. 8, pp. 292–299, Jan. 2013.
- [4] G. I. Papadimitriou, A. I. Vakali, G. Pallis, S. Petridou, and A. S. Pomportsis, "Simulation in Web Data Management," in *Applied System Simulation*, Springer US, 2003, pp. 179–199.
- [5] Z. Qu, S. Ninan, A. Almosa, K. G. Chang, S. Kuruvilla, and N. Nguyen, "Synoptic reporting in tumor pathology: Advantages of a web-based system," *Am. J. Clin. Pathol.*, vol. 127, no. 6, pp. 898–903, Jun. 2007.
- [6] C. Barry, "Issues and Perspectives on Web-based Information Systems Development," in *Third International Asia-Pacific Web Conference, 2000*.
- [7] M. Stauffer, *Laravel: Up and Running: A Framework for Building Modern PHP Apps*, 2nd ed. O'Reilly Media, 2019.
- [8] C. Pitt, *Pro PHP MVC*. Apress, 2012.
- [9] M. Miles, "Using web2py Python framework for creating data-driven web applications in the academic library," *Libr. Hi Tech*, vol. 34, no. 1, pp. 164–171, 2016.
- [10] Massimo Di Piero, "web2py for Scientific Applications," *Comput. Sci. Eng.*, vol. 13, no. 2, pp. 64–69, 2011.
- [11] I. Rauf and I. Porres, "Beyond CRUD," in *REST: From Research to Practice*, Springer New York, 2011, pp. 117–135.
- [12] R. Rodríguez-Echeverría, J. C. Preciado, J. Sierra, J. M. Conejero, and F. Sanchez-Figueroa, "AutoCRUD: Automatic generation of CRUD specifications in interaction flow modelling language," *Sci. Comput. Program.*, vol. 168, pp. 165–168, Dec. 2018.
- [13] S. P. Onesinus, "Laravel CRUD with bootstrap jQuery and Mysql," *OSPT*, 2019.
- [14] M. Kevin and K. McArthur, *Pro PHP: Patterns, Frameworks, Testing and More*, 1st ed. United States of America: Apress, 2008.
- [15] R. Rodríguez-Echeverría, J. M. Conejero, J. C. Preciado, and F. Sanchez-Figueroa, "AutoCRUD - Automating IFML Specification of CRUD Operations," in *Proceedings of the 12th International Conference on Web Information Systems and Technologies, 2016*, vol. 1, pp. 307–314.
- [16] J. Watts and G. Jorge, *CakePHP 2 Application Cookbook*. USA: Packt Publishing, 2014.
- [17] J. Martin, *Managing the database environment*. USA: Prentice Hall, 1983.
- [18] S. Tenzin, T. Lhamo, and Tsheten Dorji, "Design and Development of E-Commerce Web Application for Cooperative Store," *Int. Res. J. Eng. Technol.*, vol. 9, no. 2, pp. 846–846, 2022.
- [19] P. McFedries, *Web Coding & Development All-in-One For Dummies*. New Jersey: John Wiley & Sons, Inc., 2018.
- [20] Cake Software Foundation, *CakePHP Cookbook Documentation*. 2017.
- [21] Rails Guides Team, "Getting Started with Rails — Ruby on Rails Guides," 2015. [Online]. Available: [https://guides.rubyonrails.org/getting\\_started.html#getting-up-and-running-quickly-with-scaffolding](https://guides.rubyonrails.org/getting_started.html#getting-up-and-running-quickly-with-scaffolding). [Accessed: 28-Nov-2019].
- [22] S. Sinha, "Introduction to Laravel," in *Beginning Laravel*, Apress, 2019, pp. 1–10.
- [23] P. Noppadon and W. Panita, "Development of a Ubiquitous Learning System with Scaffolding and Problem-Based Learning Model to Enhance Problem-Solving Skills and ICT Literacy," *Int. J. e-Education, e-Business, e-Management e-Learning*, vol. 3, no. 3, 2013.
- [24] D. Golding, *Beginning CakePHP: From novice to professional*. Apress, 2008.
- [25] R. H. Hall, A. Digennaro, J. Ward, and N. Havens, "Usability Assessment Of A Web-Based Learning System For Teaching Web Development: A Progressive Scaffolding Approach," in *Information Systems, 2003*, pp. 1–10.
- [26] R. H. Mark, *Instant CakePHP Starter*. Packt Publishing, 2013.
- [27] S. I. Ahmad, T. Rana, and A. Maqbool, "A Model-Driven Framework for the Development of MVC-Based (Web) Application," *Arab. J. Sci. Eng.*, vol. 47, no. 2, pp. 1733–1747, Feb. 2022.
- [28] R. Das and L. P. Saikia, "Comparison of Procedural PHP with Codeigniter and Laravel Framework," *Int. J. Curr. Trends Eng. Res.*, vol. 2, no. 6, pp. 42–48, 2016.
- [29] R. Dāsa, "Learn CakePHP," in *Learn CakePHP*, Berkeley: Apress, 2016.
- [30] R. Hu, Z. Wang, J. Hu, J. Xu, and X. Jun, "Agile Web development with Web framework," in *2008 International Conference on Wireless Communications, Networking and Mobile Computing, 2008*.
- [31] N. Bandirmali, "mtCMF: A novel memory table based content management framework for automatic website generation," *Comput. Stand. Interfaces*, vol. 58, pp. 43–52, May 2018.
- [32] L. Daly, *Next-generation web frameworks in Python*. O'Reilly Media, Inc, 2007.
- [33] B. Kitchenham, "DESMET: A method for evaluating Software Engineering methods and tools," 1996.
- [34] S. Murugesan, "Web Application Development: Challenges And The Role Of Web Engineering," in *Web Engineering: Modelling and Implementing Web Applications*, Springer London, 2007, pp. 7–32.
- [35] M. H. Cloyd, "Designing user-centered web applications in web time," *IEEE Softw.*, vol. 18, no. 1, pp. 62–69, Jan. 2001.
- [36] M. Y. Ivory and M. A. Hearst, "Improving web site design," *IEEE Internet Comput.*, vol. 6, no. 2, pp. 56–63, 2002.
- [37] D. A. Siegel, "The business case for user-centered design," *interactions*, vol. 10, no. 3, p. 30, May 2003.
- [38] J. Rode, M. B. Rosson, and M. Perez-Quinones, "The challenges of web engineering and requirements for better tool support," *Methods*, 2002.
- [39] S. Paul, A. Mitra, and S. Dey, "Issues and challenges in web crawling for information extraction," in *Bio-Inspired Computing for Information Retrieval Applications*, IGI Global, 2017, pp. 93–121.
- [40] D. Kunda, M. Chishimba, M. Mulenga, and V. Chama, "An Analysis of Security and Performance Concerns in Mobile Web Application Development: Challenges and Open Issues," *Int. J. Recent Contrib. from Eng. Sci. IT*, vol. 5, no. 3, pp. 26–40, Oct. 2017.
- [41] P. Mole and P. V Mole, "Progressive Web Apps: A Novel Way for Cross-Platform Development," 2018.
- [42] F. Chiti, R. Fantacci, G. Pasi, and F. Tisato, "Context-Awareness in Autonomic Communication and in Accessing Web Information: Issues and Challenges," in *Wisdom Web of Things*, Springer International Publishing, 2016, pp. 107–118.
- [43] C. Krintz, "The AppScale cloud platform: Enabling portable, scalable web application deployment," *IEEE Internet Comput.*, vol. 17, no. 2, pp. 72–75, 2013.
- [44] C. Shahabi, F. Banaei-Kashani, Y. S. Chen, and D. McLeod, "Yoda: An accurate and scalable Web-based recommendation system," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 2172, pp. 418–432, 2001.
- [45] S. Murugesan and Y. Deshpande, "Meeting the challenges of web application development," in *Proceedings of the 24th international conference on Software engineering - ICSE '02*, 2002, p. 687.
- [46] A. Ginige and S. Murugesan, "The essence of Web engineering," *IEEE Multimed.*, vol. 8, no. 2, pp. 22–25, Apr. 2001.
- [47] A. W. Anuar, N. Kama, A. Azmi, and H. M. Rusli, "Analysis And Practical Application Of Web Framework And Crud Operation For Web Application Development," in *2nd International Professional Doctorate and Postgraduate Symposium 2021*, 2021, pp. 233–237.
- [48] N. Kama, S. Basri, S. A. Ismail, and R. Ibrahim, "Using static and dynamic impact analysis for effort estimation," *IET Softw.*, vol. 10, no. 4, pp. 89–95, Aug. 2016.
- [49] C. Kai, J. Omokore, and R. K. Miller, *Practical CakePHP Projects*. USA: Apress, 2009.
- [50] H. Knublauch, R. W. Ferguson, N. F. Noy, and M. A. Musen, "The Protégé OWL Plugin: An Open Development Environment for Semantic

- Web Applications,” *Int. Semant. Web Conf.*, vol. 3298, pp. 229–243, 2004.
- [51] H. V. Nguyen, C. Kästner, and T. N. Nguyen, “Exploring variability-aware execution for testing plugin-based web applications,” in *Proceedings - International Conference on Software Engineering*, 2014, no. 1, pp. 907–918.
- [52] K. Schlegel, T. Weisgerber, F. Stegmaier, M. Granitzer, and H. Kosch, “Balloon Synopsis: A JQuery Plugin to Easily Integrate the Semantic Web in a Website?,” in *Proceedings of the 2014 International Conference on Developers - Volume 1268*, 2014, pp. 19–24.
- [53] R. Connolly, R. Hoar, S. Mukherjee, and A. K. Bhattacharjee, *Fundamentals of web development*. Pearson Education, 2015.
- [54] M. Darabseh and J. P. Martins, “Risks and Opportunities for Reforming Construction with Blockchain: Bibliometric Study,” *Civ. Eng. J.*, vol. 6, no. 6, pp. 1204–1217, Jun. 2020.
- [55] D. Ko, K. Ma, S. Park, S. Kim, D. Kim, and Y. Le Traon, “API document quality for resolving deprecated APIs,” in *Proceedings - Asia-Pacific Software Engineering Conference, APSEC*, 2014, vol. 2, pp. 27–30.
- [56] A. Austin and L. Williams, “One technique is not enough: A comparison of vulnerability discovery techniques,” in *International Symposium on Empirical Software Engineering and Measurement*, 2011, pp. 97–106.
- [57] I. Medeiros, N. F. Neves, and M. Correia, “Automatic detection and correction of Web application vulnerabilities using data mining to predict false positives,” in *WWW 2014 - Proceedings of the 23rd International Conference on World Wide Web*, 2014, pp. 63–73.
- [58] Y. Mahmood, N. Kama, and A. Azmi, “A systematic review of studies on use case points and expert-based estimation of software development effort,” *J. Softw. Evol. Process*, vol. 32, no. 7, p. e2245, Jul. 2020.
- [59] S. I. Ahamed, A. Pezewski, and A. Pezewski, “Towards framework selection criteria and suitability for an application framework,” in *International Conference on Information Technology: Coding Computing, ITCC*, 2004, vol. 1, pp. 424–428.
- [60] N. Prokofyeva and V. Boltunova, “Analysis and Practical Application of PHP Frameworks in Development of Web Information Systems,” in *Procedia Computer Science*, 2016, vol. 104, pp. 51–56.
- [61] M. Giatsoglou, V. Koutsonikola, K. Stamos, A. Vakali, and C. Zigkolis, “Dynamic code generation for cultural content management,” in *Proceedings - 14th Panhellenic Conference on Informatics, PCI 2010*, 2010, pp. 21–24.
- [62] S. A. Becker and A. Berkemeyer, “Rapid application design and testing of Web usability,” *IEEE Multimed.*, vol. 9, no. 4, pp. 38–46, Oct. 2002.
- [63] Shari Thurow and Nick Musica, “Understanding Search Usability,” in *When Search Meets Web Usability, USA: New Riders*, 2009, pp. 2–14.
- [64] K. C. Haggerty and R. E. Scott, “Do, or Do Not, Make Them Think?: A Usability Study of an Academic Library Search Box,” *J. Web Librariansh.*, vol. 13, no. 4, pp. 296–310, Oct. 2019.
- [65] M. Matera, F. Rizzo, and G. T. Carughi, “Web usability: Principles and evaluation methods,” in *Web Engineering*, Springer Berlin Heidelberg, 2006, pp. 143–180.
- [66] T. Brinck and E. Hofer, “Automatically evaluating the usability of web sites,” in *CHI '02 extended abstracts on Human factors in computing systems - CHI '02*, 2002, p. 906.
- [67] Z. T. Shasha and M. Weideen, “Usability Measurement of Web-based Hotel Reservation Systems,” in *1st TESA International Conference*, 2016, no. September, pp. 1–14.
- [68] V. Thoma and J. Dodd, “Web Usability and Eyetracking,” in *Eye Movement Research*, Springer, Cham, 2019, pp. 883–927.
- [69] T. Felin and S. Kauffman, “The Search Function and Evolutionary Novelty,” *SSRN Electron. J.*, Oct. 2019.
- [70] B. Porebski, K. Przystalski, L. Nowak, and L. N. Bartosz Porebski, Karol Przystalski, *Building PHP Applications with Symphony, CakePHP, and Zend Framework*, 1st ed. GBR: Wiley Publishing, 2011.
- [71] T. Gipp and J. Ebert, “Functional web applications,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2007, vol. 4607 LNCS, pp. 194–209.
- [72] A. Gutlić and E. Mujčić, “Intelligent Web Application for Search of Restaurants and Their Services,” in *Networks and Systems*, vol. 83, Springer, 2020, pp. 452–469.
- [73] M. A. Jiménez-Crespo, “What is (not) web localization in translation studies,” *J. Int. Localization*, vol. 3, no. 1, pp. 38–60, Aug. 2016.
- [74] B. Esselink, “The Evolution of Localization,” in *Guide to Localization*, 2003, pp. 21–29.
- [75] Jiménez-Crespo and Miguel A., *Translation and web localization*. London: Routledge, 2013.
- [76] P. Zhu, “Language Problems to Be Coped with in Web Localization,” *J. Tech. Writ. Commun.*, vol. 39, no. 1, pp. 57–78, Jan. 2009.
- [77] J. Williams, “Correctly assessing the ‘ilities’ requires more than marketing hype,” *IT Prof.*, vol. 2, no. 6, pp. 65–67, Nov. 2000.
- [78] J. Hvorecký and M. Beňo, “Linkages between PISA Results and E-Working,” *Emerg. Sci. J.*, vol. 5, no. 3, pp. 294–304, Jun. 2021.
- [79] B. Tambaip, I. H. Wayangkau, S. Suwarjono, M. Loupatty, A. F. Adam, and H. Hariyanto, “Preservation of the Muyu Indigenous Language with an Android-based Dictionary,” *Emerg. Sci. J.*, vol. 4, no. 0, pp. 85–101, Oct. 2021.
- [80] S. Murugesan and A. Ginige, “Introduction and Perspectives,” in *Web Engineering, India: Idea Group Inc*, 2005.
- [81] A. W. Anuar, N. Kama, A. Azmi, and H. M. Rusli, “A Multivocal Literature Review on Records Management Potential Components in CRUD Operation for Web Application Development,” *Int. J. Model. Simulation, Sci. Comput.*, Apr. 2022.
- [82] Ira A. Penn and Gail B. Pennix, “Records inventory,” in *Records Management Handbook*, 2nd ed., New York: Routledge, 2017.
- [83] Patricia C. Franks, “Records Retention Strategies Inventory Appraisal Retention and Disposition,” in *Records and Information Management*, American Library Association, 2013, pp. 84–114.
- [84] Judith Read and Mary Lea Ginn, “Electronic Records Management,” in *Records Management*, 10th ed., Boston: Cengage Learning, 2016.
- [85] UK National Archive, “Records Management retention scheduling,” United Kingdom, 2012.
- [86] UK National Archive, “Born-digital records and metadata,” *Information Management-Digital Record Transfer*, 2017. [Online]. Available: <https://www.nationalarchives.gov.uk/information-management/management-information/digital-records-transfer/what-are-born-digital-records/>. [Accessed: 06-Dec-2020].
- [87] M. Diamond, “How to Implement a Record Retention Schedule for Electronic and Other Records,” *Association of Corporate Counsel (ACC)*, 2017. [Online]. Available: <https://www.acc.com/resource-library/how-implement-record-retention-schedule-electronic-and-other-records#>. [Accessed: 20-Feb-2021].
- [88] C. A. Lee, “Computer-Assisted Appraisal and Selection of Archival Materials,” in *Proceedings - 2018 IEEE International Conference on Big Data, Big Data 2018*, 2019, pp. 2721–2724.
- [89] R. Harvey and D. Thompson, “Automating the appraisal of digital materials,” *Libr. Hi Tech*, vol. 28, no. 2, pp. 313–322, 2010.
- [90] A. Vellino and I. Alberts, “Assisting the appraisal of e-mail records with automatic classification,” *Rec. Manag. J.*, vol. 26, no. 3, pp. 293–313, 2016.
- [91] IRMT, “Managing the Creation, Use and Disposal of Electronic Records,” *Int. Rec. Manag. Trust*, 2009.
- [92] M. Crockett, “User Guide to Retention and Disposal Schedules Council of Europe Records Management Project,” 2011.
- [93] N. W. Lo, C. Y. Wu, and Y. H. Chuang, “An authentication and authorization mechanism for long-term electronic health records management,” in *Procedia Computer Science*, 2017, vol. 111, pp. 145–153.
- [94] T. M. Masenya, “Application of modern technologies in the management of records in public libraries,” *J. South African Soc. Arch.*, vol. 53, pp. 65–79, Dec. 2020.
- [95] H. Guo, W. Li, M. Nejad, and C. C. Shen, “Access control for electronic health records with hybrid blockchain-edge architecture,” in *Proceedings - 2019 2nd IEEE International Conference on Blockchain, Blockchain 2019*, 2019, pp. 44–51.

- [96] B. Oladejo and S. Hadžidedić, "Electronic records management – a state of the art review," *Rec. Manag. J.*, vol. ahead-of-p, no. ahead-of-print, Feb. 2021.
- [97] P. Joseph, S. Debowski, and P. Goldschmidt, "Search behaviour in electronic document and records management systems: An exploratory investigation and model," *Inf. Res.*, vol. 18, no. 1, 2013.
- [98] V. L. Lemieux, "Trusting records: is Blockchain technology the answer?," *Rec. Manag. J.*, vol. 26, no. 2, pp. 110–139, Jul. 2016.
- [99] J. Namukasa, "Records management and procurement performance: A case of NAADS program in the central region of Uganda," *Rec. Manag. J.*, vol. 27, no. 3, pp. 256–274, 2017.
- [100] I. Pappel, S. Butt, I. Pappel, and D. Draheim, "On the specific role of electronic document and record management systems in enterprise integration," in *Advances in Intelligent Systems and Computing*, vol. 1184, Springer Science and Business Media Deutschland GmbH, 2021, pp. 37–51.
- [101] M. Broussard and K. Boss, "Saving Data Journalism: New strategies for archiving interactive, born-digital news," *Digit. Journal.*, vol. 6, no. 9, pp. 1206–1221, Oct. 2018.
- [102] International Council on Archives, "Principles and functional requirements for records in electronic office environments," 2013.
- [103] The National Archives United Kingdom, "Migrating information between records management systems," pp. 1–35, 2017.
- [104] L. Chen, W. K. Lee, C. C. Chang, K. K. R. Choo, and N. Zhang, "Blockchain based searchable encryption for electronic health record sharing," *Futur. Gener. Comput. Syst.*, vol. 95, pp. 420–429, Jun. 2019.
- [105] J. Lengstorf and K. Wald, *Pro PHP and jQuery*. Apress, 2016.
- [106] J. Duarte, C. F. Portela, A. Abelha, J. Machado, and M. F. Santos, "Electronic Health Record in Dermatology Service," in *ENTERprise Information Systems*, 2011, pp. 156–164.
- [107] H. Hedberg and J. Lappalainen, "A preliminary evaluation of software inspection tools, with the DESMET method," in *Proceedings - International Conference on Quality Software*, 2005, vol. 2005, pp. 45–52.
- [108] C. Marshall, P. Brereton, and B. Kitchenham, "Tools to support systematic reviews in software engineering: A feature analysis," in *ACM International Conference Proceeding Series*, 2014, pp. 1–10.
- [109] F. Faiqunisa, E. Nugroho, and P. I. Santosa, "A Model of Electronic Document Management System for Limited Partnership," *J. Telemat. Informatics*, vol. 1, no. 2, pp. 69–79, Jun. 2013.
- [110] F. D. Butterfoss, V. Francisco, and E. M. Capwell, "Choosing Effective Evaluation Methods," *Health Promot. Pract.*, vol. 1, no. 4, pp. 307–313, 2000.
- [111] E. Umamaheswari and D. K. Ghosh, "Software quality: Dual experts opinion and conditional based aggregation method," *Int. J. Eng. Technol.*, vol. 6, no. 2, pp. 1167–1175, 2014.
- [112] I. Blotenberg and A. Richter, "Validation of the QJIM: A measure of qualitative job insecurity," *Work Stress*, vol. 34, no. 4, pp. 406–417, Oct. 2020.
- [113] M. Al-aaidroos, N. Jailani, and M. Mukhtar, "Expert validation on a reference model for e-auctions that conform to Islamic trading principles," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 31, no. 1, pp. 62–71, Jan. 2019.
- [114] PMI, *A Guide to the Project Management Body of Knowledge (PMBOK Guide)*, 4th ed., vol. 40, no. 2. Project Management Institute, Inc., 2008.
- [115] J. Brooke, "SUS: A quick and dirty usability scale," United Kingdom, 1986.
- [116] J. R. Lewis, "The System Usability Scale: Past, Present, and Future," *Int. J. Hum. Comput. Interact.*, vol. 34, no. 7, pp. 577–590, Jul. 2018.
- [117] S. Ratnawati, L. Widianingsih, N. Angraini, I. Marzuki Shofi, N. Hakiem, and F. Eka M Agustin, "Evaluation of Digital Library's Usability Using the System Usability Scale Method of (A Case Study)," in *2020 8th International Conference on Cyber and IT Service Management, CITSM 2020*, 2020.
- [118] K. Finstad, "The usability metric for user experience," *Interact. Comput.*, vol. 22, no. 5, pp. 323–327, Sep. 2010.
- [119] A. Bangor, P. Kortum, and J. Miller, "Determining what individual SUS scores mean: adding an adjective rating scale," *J. usability Stud.*, vol. 4, no. 3, pp. 114–123, 2009.
- [120] S. C. Peres, T. Pham, and R. Phillips, "Validation of the system usability scale (sus): Sus in the wild," in *Proceedings of the Human Factors and Ergonomics Society*, 2013, pp. 192–196.
- [121] I. M. Pageh, A. A. J. Permana, and K. Suranata, "Usability testing and the social analysis on online counselling system for recommendations in technical vocational schools," in *Journal of Physics: Conference Series*, 2021, vol. 1810, no. 1, p. 12022.
- [122] K. Finstad, "The system usability scale and non-native English speakers," *J. Usability Stud.*, vol. 1, no. 4, pp. 185–188, 2006.
- [123] M. A. Diefenbach, N. D. Weinstein, and J. O'reilly, "Scales for assessing perceptions of health hazard susceptibility," *Health Educ. Res.*, vol. 8, no. 2, pp. 181–192, Jun. 1993.
- [124] E. P. Cox, "The Optimal Number of Response Alternatives for a Scale: A Review," *J. Mark. Res.*, vol. 17, no. 4, p. 407, Nov. 1980.
- [125] J. Nielsen, "Usability 101: Introduction to Usability," *Research-Based User Experience*, 2012. [Online]. Available: <https://www.nngroup.com/articles/usability-101-introduction-to-usability/>. [Accessed: 01-Sep-2020].
- [126] A. Holzinger, "Usability engineering methods for software developers," *Communications of the ACM*, vol. 48, no. 1, pp. 71–74, 01-Jan-2005.
- [127] J. Nielsen, "Usability metrics: Tracking interface improvements," *IEEE Softw.*, vol. 13, no. 6, pp. 12–13, 1996.
- [128] L. Faulkner, "Beyond the five-user assumption: Benefits of increased sample sizes in usability testing," in *Behavior Research Methods, Instruments, and Computers*, 2003, vol. 35, no. 3, pp. 379–383.
- [129] C. W. Turner, J. R. Lewis, and J. Nielsen, "Determining Usability Test Sample Size," 2006.