Journal of Information System and Technology Management JISTM
EISSN: 0128-1666
Volume 7 Issue 28 (November 2022) PP. 12-23
DOI: 10.35631/JISTM.728002

JOURNAL OF INFORMATION SYSTEM AND TECHNOLOGY MANAGEMENT (JISTM)
www.jistm.com

GLOBAL ACADEMIC EXCELLENCE

TOWARDS A SEMANTIC-BASED SEARCHING IN A WEB APPLICATION FRAMEWORK FOR AN IMPROVED DEVELOPMENT PRODUCTIVITY AND USABILITY

Imran Harith Azmy[1*], Azri Azmi [2], Nazri Kama[3], Hazlifah Mohd Rusli [4], Asyraf Wahi Anuar[5]

[1] Razak Faculty of Technology and Informatics, Universiti Teknologi Malaysia
Email: imranharithazmy@graduate.utm.my
[2] Razak Faculty of Technology and Informatics, Universiti Teknologi Malaysia
Email: azriazmi@utm.my
[3] Razak Faculty of Technology and Informatics, Universiti Teknologi Malaysia
Email: mdnazri@utm.my
[4] Razak Faculty of Technology and Informatics, Universiti Teknologi Malaysia
Email: hazlifah@utm.my
[5] Razak Faculty of Technology and Informatics, Universiti Teknologi Malaysia
Email: asyraf.wahi@gmail.com
[*] Corresponding Author

Article Info:

Article history:
Received date: 01.08.2022
Revised date: 30.08.2022
Accepted date: 25.09.2022
Published date: 09.11.2022

To cite this document:

Azmy, I. H., Azmi, A., Kama, N., Rusli, H. M., & Anuar, A. W. (2022). Towards a Semantic-based Searching in a Web Application Framework for an Improved Development Productivity and Usability. *Journal of Information System and Technology Management, 7* (28), 12-23.

DOI: 10.35631/JISTM.728002

Abstract:

The increasing usage and advancements of Web Application technologies have resulted in the demand for quick development of high-quality web applications. The situation resulted to the emergence of various frameworks and methods for a rapid application development of web applications, such as code generators. Conventional code generators generally anchor basic and generic operations of create, read, update and delete (CRUD) for efficient development through automation by eliminating redundancy and unnecessary work during coding. However, as technologies and use cases of web applications progress, information retrieval or 'search' became one of the most common features of a web application apart from the very basic CRUD-related functions. A literature review process and analysis of existing web application console framework has been done towards the proposal of solution. We proposed incorporating the search feature into code generators for rapid web application development of the CakePHP framework. Hence, an architecture of the new framework has been presented in this study for future developments that aims to enhance usability and efficient software developments.

Keywords:

Information Retrieval, CakePHP, Metadata, Web Application, Rapid Application Development, Usability, Search Engine

Introduction

Web application development is critical in the current rapid worldwide growth of industry since most organizations have their own web apps to satisfy the needs of their different enterprises. Different needs result in varying degrees of complication, posing a fresh barrier for Web application development. The risk of being unable to deliver the application on time and to the expected standard is increasing (Lee et al., 2017). As new web technologies keep on emerging, it is clear that this trend will continue. Software engineering development of Web application systems are expected to increase continuously, eventually accounting for the great majority of software project development (Luo, 2021). In this high-paced electronic environment, searching via a search engine is one of the most important methods to find and retrieve information. For public-consumed web applications such as e-commerce sites or academic depositories, efficient searching allows maximum outputs for its users (López & Cuadrado, 2020). However, traditional searching engines for information retrieval may no longer be productive in meeting today's expectations, especially in understanding the meaning of the query (Kumar & Singh, 2017). Moreover, basic systems generally have great difficulty adapting to dynamic changes on a webpage (Ramalingam et al., 2022). Hence, a technology that enables the machine to understand the searching or information retrieval context further may solve this issue. Numerous meta or semantic-based search engines with various and efficient result-providing characteristics are deployed to obtain the best results for queries (Kumar & Singh, 2017; Lechtenberg et al., 2022; Ramalingam et al., 2022; Tamine & Goeuriot, 2022). Therefore, there is a severe need to introduce an improved version of search engines for information retrieval for the development of any common types of web applications.

In conjunction with the mentioned issue, we look into Rapid Application Development (RAD) methods whereby code generator is a popular instrument for developing web applications. Half-baked generated codes of common operations in web applications, by rapid application development tools or frameworks are always being mitigated by additional enhancements, human code modifications, and functional integration, which is time-consuming and involves repetitive coding (Rodriguez-Echeverria et al., 2018). Supposedly, a more productive and effective application development takes place by minimising customisations and modifications after deployment, and hence minimising risks. Concerning the challenge as mentioned earlier, the development and integration of a search function using the code generation method are likely to encounter that very same issue, particularly the additions and modifications required to produce better results for queries, as previously described. The better solution is to enhance the search function beforehand, at the pre-generation of codes stage. Substituting a traditional search engine to search engines that produce better retrieval results into code generators of rapid web application development methods may be an effective solution. In addition, studies on this issue have been lesser-known even though it is a significant matter. Therefore, this paper intends to enrich traditional information retrieval mechanisms and their methods of development by introducing a metadata or semantic-based search engine as an integral feature of the software product for an improved version of code generator instruments of RAD methods towards better usability and productivity.

This research intends to establish and move towards a new paradigm for rapid web application development through CRUD code generation that is equipped with semantic based information retrieval capabilities, with the main goal of swiftly building high-quality web applications to fulfil the demand created by the rise of web-based technologies. Web applications have been developed based on the traditional CRUD operation, which is believed needs to be improvised due to the current complexities of web applications for quicker and simpler development. The combination of the proposed improvised CRUD operation model with RAD principles of software development through the enablement of automation and scaffolding, therefore this study would be revolutionary for the web application development world, especially in terms on how easily high-quality web applications can be developed. Ultimately, it is hoped that the proposed solution would be emerged as a powerful RAD solution, especially within the web application development community, in the future.

The paper outlines related works in section 2, literature reviews in section 3, the proposed solution in section 4 and finally, the conclusion in section 5. It shall be noted that the proposed solution outlined in this paper is an initial work for a series of research and developments.

Related Works

This section explains the works other researchers did that initially inspired us to conduct this study, which later expands into literature review for an in-depth understanding of the topic and issue to formulate our proposed solution.

The main inspiration and reference for our study is based on the works from authors of (Anuar et al., 2022) which introduced the Re-CRUD. They have developed an automation framework for code generation is a vital reference for our study. The search feature has been highlighted as one of the important feature to be proposed to improve conventional web application framework which is limited to CRUD operations as its fundamental functions. They emphasised that further modifications are often required to include other useful features, which search is one of them.

Moreover, authors of (Funk et al., 2016) have been developing an application for social media and speech conversation-related use cases. The developed application offers five primary operations throughout its architecture; the conventional essential create, read, update and delete, together with the search operation called 'SCRUD'. This software is described as a novel approach to storing, analysing, and exploiting various types of conversational data that have been validated in use with a successful software deployment that includes several working input and output models (Funk et al., 2016). Another work is by (Karnouskos et al., 2013), in which a similar combination Search + CRUD (SCRUD) operations has been the basic pattern of developing the user and asset management modules within a Smart Grid neighborhood web application that has been developed as part of the study (Karnouskos et al., 2013).

CRUD operations are typically performed over ontological repositories with search operations in applications that leverage Semantic Web technologies (Bartalos, 2014). A method for carrying out these operations (CRUD and Search) based on a semi-automatic form generation and automatic object ontology mapping with bean generation was presented. The method was tested in two domains: the online labour market and scientific publication within a portal that provides information on a specific topic. The proposed method is said to be suitable for a wide range of Semantic Web applications that require persistent storage of data entities (Bartalos, 2014). Therefore, the search operation is certainly one of the vital components if the conventional CRUD operation shall be improvised through the combination of a new operation.

Meanwhile, authors of (A. Madhavi, 2013) studied the characteristics and architectures of Meta Search Engines (MSE) for extracting documents from one or more web domains that focused on general and special purpose MSE. To obtain the relevant search results, the authors of presented an MSE based on clustering and ranking. Its core modules are a user interface, a relevancy calculation, a cluster generator, and a webpage adjuster (Chandakanna & Vatsavayi, 2014). An MSE employs self-forward query interfaces to SEs to produce their findings. MSE takes the user's input and sends it to many SEs simultaneously to produce results. The acquired result is then formatted and presented to the user. (Mariappan et al., 2013) proposes using a set of keywords in a single query and the Word Net ontology to offer the most appropriate question to SE via MSE. Therefore, it is also important to consider the features of MSEs in our study.

Therefore, the related works, as explained, have provided us with the fundamentals and basic grounds of this study, which leads into the formulation of the proposed solution. The next section outlines a literature review for an in-depth understanding of the topic and issue.

Literature Review

This chapter outlines the literature reviews conducted within this study, essentially based on the vital related topics from the established problem identification, which has been derived into its proposal of solution. This section aims to better understand the issues according to specific topics and contexts based on existing studies and works done by other researchers and practitioners.

Web Application Development

Web applications drove us into the information age and have since become essential to our daily lives. This trend will continue when new web technologies emerge. Web application systems are predicted to grow, eventually accounting for the vast majority of software project developments (Luo, 2021). Many traditional databases and information systems are being migrated to the Internet. A large number of complex distributed applications are gaining traction on the Web. The Web has grown in popularity and ubiquity due to its ability to provide publishing information that supports all types of content linkages and is easy to access by end-users. Hence, speedy, highly efficient, and reliable web application development methodologies must follow both general and specific requirements and use cases (Luo, 2021).

To ensure smooth, quick and systematic web application development, a web application framework is conventionally used. A web application framework is a software development framework created specifically to help developers create web-based applications. A web application framework's purpose is to make it quicker and easier to create secured web applications in a timely and effective manner (Laaziri et al., 2019). The most challenging task is deciding on the best framework for the web application. Since there are so many different types of frameworks, it's not an easy decision. Furthermore, choosing the wrong framework can negatively affect the web application. As the web has grown, many frameworks have emerged, including PHP, Java servlets, struts, Stencil, Rebel, Ruby on Rails, Flask, and Falcon, to facilitate various factors and situations (Curie et al., 2019). Web applications typically send and retrieve large amounts of data daily. Therefore, choosing a programming framework is critical when developing web applications. However, alternatives to conventional web application programming frameworks for developing a web application have emerged, according to various studies and practices, which will be thoroughly discussed throughout the following sections.

Rapid Web Application Development

The release of a text by James Martin with the same title appears to have sparked interest in RAD (Martin, 1983). According to Martin, RAD's main goals are high-quality systems, quick development and delivery, and minimal costs. These goals can be summed up as: the commercial requirement to produce working business applications in less time and for less cost. Since the software product should always be delivered speedily, Rapid Application Development (RAD) has been a crucial paradigm for web application developments. To save development time and costs, RAD uses automated tools and methodologies by prioritising rapid prototype releases and iterations (Naz & Khan, 2015; Pissinati et al., 2019). The approach of constructing and releasing prototypes aims to jumpstart the application development and as a process for proof of concepts for clients or stakeholders (Pissinati et al., 2019).

RAD combines multiple structured methodologies with prototype and collaborative application development techniques to speed the development of applications. Requirements Planning Phase, User Design Phase, Construction Phase, and Cut-Over Phase are the four general iterative steps of Rapid Application Development (Qodim et al., 2019). RAD is also described as a group of approaches that grew in reaction to the flaws and variants of waterfall development, in which special techniques and computer tools are used in RAD to speed up the analysis, design, and implementation phases to get a component of the system produced and into the hands of users for assessment and feedback as rapidly as possible. RAD has four essential components, which are methodology, people, management, and tool, which determine the effectiveness of the software development process (Dennis et al., 2012).

Code Generators for Rapid Web Application Development

Tool is one of the essential pillars of RAD, and there are many tools that significantly aid software development that have been widely used in practice. CASE (computer-assisted software engineering) tools, JAD (joint application development) sessions, fourth-generation/visual programming languages, and code generators are all essential examples of tools within the RAD paradigm (Dennis et al., 2012). As 'tool' is one of the fundamental pillars of RAD, numerous tools have been widely utilised in practise to greatly improve software development. Automatic code generators are a prominent tool within the RAD essential components, which is used to reduce the difficulty of writing common capabilities inside the application, such as software functionalities that are made up based on the CRUD operations.

Based on the "Develop Once, Run Everywhere" principle, a technique for automatic code generation for application development based on the Model Driven Architecture is suggested by (Benouda et al., 2018). Its algorithm handles all CRUD operations, smoothing and speeding up the development process with the goal of making application development easier and faster. By expressing a solution in high-level abstractions and using models to produce the common parts of applications, they have proved the potential of model-driven techniques to boost development productivity and improve application quality (Benouda et al., 2018).

Moreover, a model-driven framework for cross-platform applications, web and mobile, that generates scaffolding codes for CRUD operations features has been a promising solution (Inayatullah et al., 2019). The Model Driven Architecture (MDA) and "Develop once, run everywhere" principles are used to build the framework. A UML-based Domain-Specific Modelling Language (DSML) with multiple data types and stereotypes has been developed. A comprehensive open source transformation engine has been built as part of the study to automatically produce scaffolding CRUD (Create, Read, Update, and Delete) code for cross-platform applications. A comprehensive open-source transformation engine can be built to automatically generate scaffolding CRUD code for cross-platform applications from high-level models, such as the UML class diagram (Inayatullah et al., 2019). A content management framework for automatic website generation that can provide Application Programming Interfaces (APIs) and are flexible enough to meet unique business and user requirements may also be a promising solution. An adaptive scaffolding architecture could be implemented to generate Create, Read, Update, and Delete (CRUD) interfaces for all sorts of database tables (Bandirmali, 2018).

Search Feature

Search, also known as information retrieval, is an integral part of the information-gathering process. Its roots can be traced back to database searching when a user wanted to find various resources from a larger pool of possibilities (Joseph et al., 2013). Search feature in web applications basically is a function that allows users to find for and retrieve records using words or phrases. It's critical in any WA because it allows quick data retrieval using the search criteria (Joseph et al., 2013). Generally, a user performs a query on the Search Interface to obtain information and expects the best possible result. The search engine responds by displaying a list of web pages containing the requested information (Gupta et al., 2014). When search engines are utilised to execute the same activity, the variety of results and coverage grows, which can improve user satisfaction (Kumar & Singh, 2017; Liu et al., 2018; Vila et al., 2021; Zhang et al., 2018). Nonetheless, (Joseph et al., 2013) highlighted that there is a low understanding for users' search behaviour.

Metadata is crucial in the digitisation of indexing and cataloguing. Metadata aids in the organisation of electronic records, facilitates system interoperability, provides digital identification, and aids in archiving and preservation (Radzuan et al., 2018). Therefore, metadata is crucial in digital environments, especially web applications that typically have search or information retrieval features, and its potential benefits should not be ignored.

Moreover, according to (Mariappan et al., 2013), to improve conventional search engines, meta-search engines were created whereby the efficiency of web searches by expanding the number of indexed results is improved. To retrieve the most relevant results, the query should be more refined, detailed, and comprehensive (Mariappan et al., 2013). Another study by (Malhotra & Rishi, 2021) has developed a new intelligent metadata-based search engine that fuses data from diverse data collections, such as their specialised domains and information kinds for E-commerce use cases. Metadata from various sources is combined to create dynamic controls in which all critical parameters expected from next-generation large data processing systems, such as scalability, partial failure support, and extensibility, are well satisfied by the suggested technique. An exhaustive and complete experimental examination demonstrates the efficiency and efficacy.

A study conducted by (Mahajani et al., 2019) has resulted to a development of a meta-search engine result optimization system that incorporates both linear and semantic search approaches. The proposed approach, according to the experimental results, generates a well-optimized search result list. Hence, it could be a vital consideration for enhancing the proposed framework. On the other hand, (A. Madhavi, 2013) emphasised that the selection of the source engine and its integration have been the focus of research-related metadata-based search engines. Clustering and linguistic analysis, which seek to show themes within results, as well as textual analysis and display, which can help dig deeper into a set of results, are examples of recent smarter searching technologies (A. Madhavi, 2013).

Based on the literatures reviewed, we observe numerous remarkable advancements of metadata and semantics-based approaches for information retrieval have been developed for other use cases. We foresee these developments as opportunities that could be adapted into RAD methodologies for web application development. Hence, these works are very useful for formulating the proposed solution and its future works within our study. The detailed discussion on the research method is explained in the following section.

Research Methods

This section explains the research method for this study which ultimately result to the proposed solution, that will be discussed in the next section. Figure 1 illustrates the research method for this study.

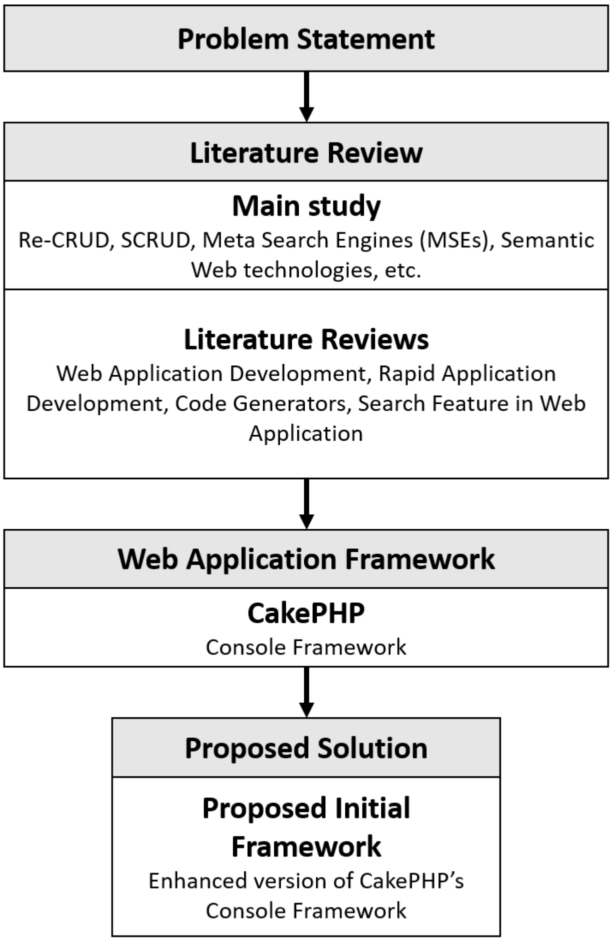
Problem Statement
Literature Review
Main study
Re-CRUD, SCRUD, Meta Search Engines (MSEs), Semantic Web technologies, etc.
Literature Reviews
Web Application Development, Rapid Application Development, Code Generators, Search Feature in Web Application
Web Application Framework
CakePHP
Console Framework
Proposed Solution
Proposed Initial Framework
Enhanced version of CakePHP's Console Framework

Figure 1: Research Method

This study begins with the formulation of the problem statement, which finally aims to to improve usability and development productivity through an enhanced version of code generator by enriching its information retrieval mechanisms and development methodology. This may be done by introducing a semantic-based engine as the integral feature of the software product using the enhanced code generator. A literature review process has been done to investigate further regarding the mentioned idea, whereby main and secondary studies has been identified. The main studies include works done by other researchers that are related to CRUD generators such as Re-CRUD and SCRUD, Meta Search Engines (MSEs) and Semantic Web Technologies. Meanwhile, secondary studies are meant to support the the main studies towards proposing the solution have been reviewed. Next, we take a look into CakePHP, a popular web application development framework. Within CakePHP, we analysed its console framework, a CRUD code generator mechanism which we will propose to improvise towards our main purpose of this study to ultimately propose an enhanced version of the code generator framework. The proposed solution, which is our current and initial proposal for this study, is explained in the following section.

Proposed Solution

This section explains the proposed solution for the problem discussed earlier. The proposed solution is outlined in the form of a conceptual framework which refers to the overall logical orientation and associations of everything that comprises the underlying thinking, structures and plans of the entire research (Kivunja, 2018). This proposed framework is our initial framework at this point, with plans for future enhancements based on projected extensive research and development. Future works planned as the continuation of this initial study are outlined in the Conclusion section.

In this section, we present the proposed framework based on the problems defined earlier in which the search function is integrated as the main component for developing the web application through a code generator framework, or also known as the console framework of CakePHP (CakePHP, 2020), which originated from the CRUD Model proposed (Martin, 1983). This framework is highly influenced by (Anuar et al., 2022) that proposed the Re-CRUD in which they proposed a code generator framework with an additional eleven features on top of CRUD functionalities. However, from our awareness, an in-depth study into the 'Search' functionality has not been discussed by the authors of (Anuar et al., 2022). Hence, this study emphasised and focussed on the 'Search' functionality as the most essential feature that needs to be prioritised for detailed research and finally formulating the proposed framework as the initial solution.

The proposed framework is illustrated within the system architecture of a typical application that is developed and running on CakePHP, connected to the httpd application server, and interfaced by a web browser via intranet networks. As mentioned, the search function is integrated as the main component for developing the web application through a code generator framework, also known as CakePHP console framework (CakePHP, 2020). The pre-generated search function shall have two main features; a decent user-friendly interface on the front-end and a semantic/metadata-based search engine. Figure 2 illustrates the proposed initial framework.

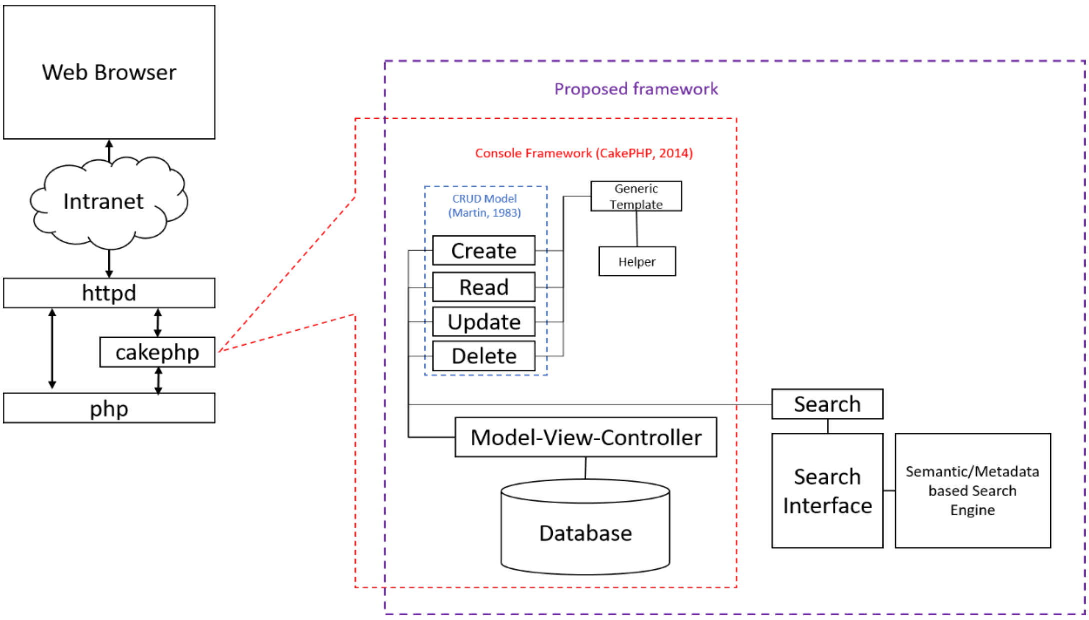

Figure 2: Proposed Code Generator Framework for Rapid Web Application Development (Initial Framework)

On top of that, the search mechanism that shall be integrated into the framework should be equipped with a semantic or metadata-based search engine. However, at this stage, we have not decided on the details of the search engine and interface but will be further discussed in the next section (Conclusion). Most importantly, the search engine and interface should have the following characteristics, in Table 1.

Table 1: Characteristics for The Search Engine of The New Code Generator Framework

Characteristic	Explanation
Easy to integrate	Integration of the developed or open-source search engine into the code generator should not be complex.
Highly usable	The search engine should be functional upon deployment for user satisfaction.
Effective searching results	Searching results should be acceptably accurate based on user inputs through the search interface.

Conclusion

Traditional code generators typically anchor simple and generic activities such as creation, read, update, and delete (CRUD) for faster development by eliminating repetition and unnecessary coding works. Apart from the most basic CRUD-related operations, information retrieval or 'search' has become one of the most popular features of a web application as technologies and use cases of web applications have progressed. The importance of this study is generally to provide a new paradigm of web application development through CRUD code generation that is equipped with semantic-based searching capabilities, with the main idea of developing quality web application rapidly to meet demands due to the emergence of web-based technologies. We proposed that the CakePHP framework's code generators include a search capability for rapid web application development. As a result, a new framework's architecture has been presented in this study for future developments with the goal of improving usability and efficiency. Based on the discussions outlined throughout this study, we strongly intend to continue carrying out the following works for deeper understanding, knowledge and a more comprehensive solution related to the topic. In the future, we intend to identify open-source search engines or algorithms; to be integrated into the framework and finally test its performance; search results relevancy/accuracy for further enhancement of the framework.

Acknowledgement

The study is financially supported by the Fundamental Research Grant Scheme (Vote No. R.K130000.7856.5F415) awarded by the Ministry of Higher Education of Malaysia (MOHE) and Universiti Teknologi Malaysia.

References

A. Madhavi, K. H. C. (2013). Architecture Based Study Of Search Engines And Meta Search Engines For Information Retrieval. *International Journal of Engineering Research & Technology (IJERT)*, *2*(5), 1832–1835.

Anuar, A. W., Kama, N., Azmi, A., Rusli, H. M., & Yahya, Y. (2022). *Re-CRUD Code Automation Framework Evaluation using DESMET Feature Analysis*. *13*(5), 437–452.

Bandirmali, N. (2018). mtCMF: A novel memory table based content management framework for automatic website generation. *Computer Standards and Interfaces*, *58*, 43–52. https://doi.org/10.1016/j.csi.2017.12.002

Bartalos, P. (2014). *(S) CRUD Pattern Support for Semantic Web Applications*. *221512840*.

Benouda, H., Azizi, M., Moussaoui, M., & Esbai, R. (2018). Automatic code generation within MDA approach for cross-platform mobiles apps. *Proceedings of EDIS 2017 - 1st International Conference on Embedded and Distributed Systems*, *2017-Decem*, 1–5. https://doi.org/10.1109/EDIS.2017.8284045

CakePHP. (2020). CakePHP Cookbook Documentation. In *CakePHP 4* (4th ed.). Cake Software Foundation. https://book.cakephp.org/4/_downloads/en/CakePHPCookbook.pdf

Chandakanna, V. R., & Vatsavayi, V. K. (2014). A model view controller based Self-Adjusting Clustering Framework. *Journal of Systems and Software*, *89*(1), 193–206. https://doi.org/10.1016/j.jss.2013.11.1112

Curie, D., Jaison, J., Yadav, J., & Fiona, J. (2019). Analysis on Web Frameworks. *Journal of Physics: Conference Series*, *1362*, 12114. https://doi.org/10.1088/1742-6596/1362/1/012114

Dennis, A., Wixom, B. H., & Roth, R. M. (2012). *Systems Analysis & Design*. John Wiley & Sons, Inc.

Funk, A., Gaizauskas, R., & Favre, B. (2016). *A Document Repository for Social Media and Speech Conversations*. https://aclanthology.org/L16-1070/

Gupta, S., Kanakam, P., & Vishnu, S. M. H. (2014). An Analysis of Exploring Information from Search Engines in Semantic Manner. *International Journal of Advanced Research in Computer Science and Software Engineering*, *4*(5), 2277. www.ijarcsse.com

Inayatullah, M., Azam, F., & Anwar, M. W. (2019). Model-Based Scaffolding Code Generation for Cross-Platform Applications. *2019 IEEE 10th Annual Information*

Technology, Electronics and Mobile Communication Conference, IEMCON 2019, 1006–1012. https://doi.org/10.1109/IEMCON.2019.8936289

Joseph, P., Debowski, S., & Goldschmidt, P. (2013). Search behaviour in electronic document and records management systems: An exploratory investigation and model. *Information Research*, *18*(1), 1–31.

Karnouskos, S., Da Silva, P. G., & Ilić, D. (2013). Developing a web application for monitoring and management of Smart Grid neighborhoods. *IEEE International Conference on Industrial Informatics (INDIN)*, 408–413. https://doi.org/10.1109/INDIN.2013.6622919

Kivunja, C. (2018). Distinguishing between theory, theoretical framework, and conceptual framework: A systematic review of lessons from the field. *International Journal of Higher Education*, *7*(6), 44–53. https://doi.org/10.5430/ijhe.v7n6p44

Kumar, N., & Singh, P. (2017). Meta Search Engine with Semantic Analysis and Query Processing. *International Journal of Computational Intelligence Research*, *13*(8), 2005–2013. https://doi.org/10.37622/ijcir/13.8.2017.2005-2013

Laaziri, M., Benmoussa, K., Khoulji, S., & Kerkeb, M. L. (2019). A Comparative study of PHP frameworks performance. *Procedia Manufacturing*, *32*, 864–871. https://doi.org/10.1016/j.promfg.2019.02.295

Lechtenberg, F., Farreres, J., Galvan-Cara, A. L., Somoza-Tornos, A., Espuña, A., & Graells, M. (2022). Information retrieval from scientific abstract and citation databases: A query-by-documents approach based on Monte-Carlo sampling. *Expert Systems with Applications*, *199*(March), 116967. https://doi.org/10.1016/j.eswa.2022.116967

Lee, T. K., Wei, K. T., & Ghani, A. A. A. (2017). Systematic literature review on effort estimation for Open Sources (OSS) web application development. *FTC 2016 - Proceedings of Future Technologies Conference, December*, 1158–1167. https://doi.org/10.1109/FTC.2016.7821748

Liu, M., Liu, Y., Mao, J., Luo, C., Zhang, M., & Ma, S. (2018). "satisfaction with Failure" or "unsatisfied Success": Investigating the Relationship between Search Success and User Satisfaction. *The Web Conference 2018 - Proceedings of the World Wide Web Conference, WWW 2018*, *2*, 1533–1542. https://doi.org/10.1145/3178876.3186065

López, J. A. H., & Cuadrado, J. S. (2020). MAR: A structure-based search engine for models. *Proceedings - 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems, MODELS 2020*, 57–67. https://doi.org/10.1145/3365438.3410947

Luo, L. (2021). Web application software engineering technology and process. *Proceedings of IEEE Asia-Pacific Conference on Image Processing, Electronics and Computers, IPEC 2021*, 935–937. https://doi.org/10.1109/IPEC51340.2021.9421250

Mahajani, A., Pandya, V., Maria, I., & Sharma, D. (2019). *A Novel Approach for Meta-Search Engine Optimisation*. *904*(July), 339–351. https://doi.org/10.1007/978-981-13-5934-7

Malhotra, D., & Rishi, O. (2021). An intelligent approach to design of E-Commerce metasearch and ranking system using next-generation big data analytics. *Journal of King Saud University - Computer and Information Sciences*, *33*(2), 183–194. https://doi.org/10.1016/j.jksuci.2018.02.015

Mariappan, A. K., Suresh And, R. M., & Subbiah Bharathi, V. (2013). Semantic meta search engine using semantic similarity measure. *Journal of Theoretical and Applied Information Technology*, *50*(3), 523–532. https://doi.org/10.24924/ijise/2015.11/v3.iss2/73.79

Martin, J. (1983). *Managing the Data-base Environment*. Prentice-Hall.

Naz, R., & Khan, M. N. A. (2015). Rapid applications development techniques: A critical review. *International Journal of Software Engineering and Its Applications*, *9*(11), 163–176. https://doi.org/10.14257/ijseia.2015.9.11.15

Pissinati, P. de S. C., Évora, Y. D. M., Rossaneis, M. A., Gvozd, R., Dos Santos, M. S., & Haddad, M. D. C. F. L. (2019). Development of a web software prototype to support retirement planning. *Revista Latino-Americana de Enfermagem*, *27*. https://doi.org/10.1590/1518-8345.3024.3169

Qodim, H., Busro, & Rahim, R. (2019). Islamic Calendar: Prototype of Hijri Calendar Application using Rapid Application Development Method. *2019 7th International Conference on Cyber and IT Service Management, CITSM 2019*, 5–8. https://doi.org/10.1109/CITSM47753.2019.8965410

Radzuan, W. E. U. M., Yatin, S. F. M., Junaide, N. A. S. R., & Mazlan, N. (2018). Indexing and cataloguing in Digital Environment. *International Journal of Academic Research in Business and Social Sciences*, *8*(9), 202–213. https://doi.org/10.6007/ijarbss/v8-i9/4585

Ramalingam, M., Saranya, D., Shankarram, R., Chinnasamy, P., Ramprathap, K., & Kalaiarasi, A. (2022). An Automated Framework For Dynamic Web Information Retrieval Using Deep Learning. *2022 International Conference on Computer Communication and Informatics, ICCCI 2022*. https://doi.org/10.1109/ICCCI54379.2022.9741044

Rodriguez-Echeverria, R., Preciado, J. C., Sierra, J., Conejero, J. M., & Sanchez-Figueroa, F. (2018). AutoCRUD: Automatic generation of CRUD specifications in interaction flow modelling language. *Science of Computer Programming*, *168*, 165–168. https://doi.org/10.1016/j.scico.2018.09.004

Tamine, L., & Goeuriot, L. (2022). Semantic Information Retrieval on Medical Texts: Research Challenges, Survey, and Open Issues. *ACM Computing Surveys*, *54*(7). https://doi.org/10.1145/3462476

Vila, T. D., González, E. A., Vila, N. A., & Brea, J. A. F. (2021). Indicators of website features in the user experience of e-tourism search and metasearch engines. *Journal of Theoretical and Applied Electronic Commerce Research*, *16*(1), 18–36. https://doi.org/10.4067/S0718-18762021000100103

Zhang, F., Zhou, K., Shao, Y., Luo, C., Zhang, M., & Ma, S. (2018). How well do offline and online evaluation metrics measure user satisfaction in web image search? *41st International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2018*, *3*, 615–624. https://doi.org/10.1145/3209978.3210059