CEPAT Journal of Computer Engineering: Progress, Application and Technology
Vol. 1, No. 3, November 2022, pp. 18~27
ISSN: 2963-6728, DOI: https://doi.org/10.25124/cepat.v1i03.5297

Content management system development for medical web applications

Monterico Adrian[1] and Munawar Ahmad Z.A[1]
[1]School of Computing, Telkom University, Indonesia
[2]Bandung Institute of Technology, Indonesia

Article Info

Article history:

Received August 14, 2022
Revised August 29, 2022
Accepted August 31, 2022

Keywords:

web app
Content management system
Medical

ABSTRACT

The world of the web and the internet today has become one of the centers for exchanging information between humans, one of which is shown by the development of web applications in the form of content management systems (CMS). By being based on JavaServer Pages (JSP), which has the advantages of being platform-independent, inexpensive and free, and easy to develop, it is possible to become an alternative solution to meet today's general medical needs, namely online and web-based, extensible and customizable. , robust and efficient, secure and authenticated, and easy to develop and maintain. There is a need that turns out to be equally owned and used by only a few medical institutions, or there is even a need that is shared by all of them. These needs are developed and used as a standard so that web applications in the medical field are universal and can be used by any medical institution, anywhere, and anywhere. As a result, medical needs can be implemented into a web application and universally applicable.

Corresponding Author:

Monterico Adrian
School of Computing, Telkom University
Bandung, Indonesia
Email: monterico@telkomuniversity.ac.id

1. INTRODUCTION

The world of the web and the internet today has become one of the backbones and centers for exchanging information between humans. This is due to the special characteristics of the web and the internet, which are globally accessible, cheap and fast costs, anonymity, and weak regulations in regulating them. Coupled with other advantages of the web and the internet in terms of functionality, flexibility, and ease of use, one of which is indicated by the content management system (CMS), which is an application that allows someone to manage and/or manipulate the content of a website [1], [2]. With this CMS, the creation of websites, portals, and other web and internet applications can be done relatively easily and efficiently so that the delivery of information becomes faster and more effective.

The medical world today has become an important element for humans. This is caused by human factors that cannot be separated from illness and need treatment to meet their health needs. But in the process, often the people responsible for meeting this need, namely doctors, nurses, and other medical-related jobs, make small or big mistakes, which are caused by human factors as well, which will never be free from mistakes. Coupled with the high risk in the medical field, the extreme case is related to a person's life. Therefore, a need was born for a system that can help humans in this regard, namely, a system that can reduce the error rate and minimize the possible risks.

Implementing information technology (IT) can be found in many fields. Gong et al. developed a Hospital information system using cloud computing [3], [4]. The use of sensors and machine learning techniques in mining can be found in [5], [6][7], [8]. In the renewable energy field, Ahmed et al report the use of Solar energy for water desalination [9], [10]. In healthcare, Bhavsar et al reviews the machine learning

techniques for medical diagnosis [11][12]. In addition to that, artificial intelligence can also be used to classify and detect skin cancers [13]. In climatology, IT can be used for wind speed prediction [14]. In geophysics, Mousavi et al. use a machine learning approach for earthquake magnitude estimation [15], [16]. Therefore, it is challenging to develop web-based applications for medical purposes.

Software in the form of a web application, namely a CMS, can be an alternative solution for these needs and phenomena. This is supported by the factor of this software that can meet the general medical needs mentioned earlier, namely online and web-based, extensible and customizable, robust and efficient, secure and authenticated, and coupled with advantages in ease of development and maintenance so that it is suitable and feasible to be applied in any medical institution, which will indirectly improve the quality of medical services from the medical institution.

The concept of web applications was introduced in 1999 in the Java programming language in the Servlet Specification version 2.2. At that time, JavaScript and eXtensible Markup Language (XML) had been developed but not yet for Asynchronous JavaScript and XML (AJAX). The XMLHttpRequest object was recently introduced in Internet Explorer 5 as an ActiveX object. In software engineering, a web application is an application that is accessed using a web browser over a network such as an internet or an intranet. A web application is also a computer software application coded in a language the web browser supports, such as HyperText Markup Language (HTML), JavaScript, AJAX, Java, and others. It relies on that browser to display the application. Web applications became famous because of the easy availability of client applications to access them, web browsers, which are sometimes referred to as thin clients. The ability to update and maintain web applications without having to distribute and install software on thousands of client computers is a crucial reason for its popularity [17], [18].

A Content Management System (CMS) is an application or software that allows someone to manage or manipulate a website's content. Generally, a CMS consists of two elements: a content management application (CMA) and a content delivery application (CDA). The CMA element allows content managers who may not have knowledge of HTML, to manage the creation, modification, and deletion of content from a website without needing webmaster skills. At the same time, the CDA element uses and collects information previously added, removed, or changed by the website owner to update or update the website.

The primary explanation of the CMS [19] works is as follows. Content is added and saved through the user interface. Web pages are generated and published. The URL of the web page is listed in the navigation menu of the website. All index pages and navigation menus are updated automatically. The problem formulated in this research is to know the needs in the medical field that can be implemented into a web application and can be universally applied to all medical institutions. Meanwhile, the research objective to be achieved in carrying out this research is to explore and find out what needs in the medical field can be implemented in a web application and can be universally applicable to all medical institutions.

The systematics of writing this research report are as follows. The introduction explains the background, problem formulation, objectives, problem boundaries, methodology, and systematic discussion used to compile research reports. The chapter also briefly discusses the theoretical basis used in research analysis, design, and implementation. The Methods section contains a solution analysis of medical problems and needs that can be implemented and universally applicable, which will be used as the basis for the following software analysis and design phase. In addition, this section describes the analysis and design of the software that will be used as the basis for the implementation phase to be carried out next. The results and discussion section contains the implementation of the software's analysis and design and testing software. The conclusion section contains the conclusions and suggestions obtained during the performance of the study.

2. METHOD

This section contains various theoretical foundations. This will discuss the definition of software development methods used, namely Unified Markup Language (UML) and Unified Process, and theoretical approaches to implementing medical needs.

2.1 UML

UML 2.0 divides diagrams into two categories: structural diagrams and behavioral diagrams. Structural diagrams describe the physical organization of an object in the system. For example, how one object relates to another object. There are several structural diagrams in UML 2.0, namely [20], [21]:

1. Class Diagram

This diagram uses classes to describe the entities that make up the system and the static relationships between these entities. Class diagrams are one of the most commonly used UML diagrams.

2. Component Diagram

This diagram shows the organization's role in the implementation of a system. These diagrams can group smaller elements, such as classes, into larger sections.

3. Composite Structure Diagram

This is a new diagram in UML 2.0. As the system becomes more complex, the relationships between elements also become more complex. Conceptually, this diagram connects class diagrams with component diagrams. This diagram shows how elements in the system combine to realize complex patterns.

4. Deployment Chart

This diagram shows how a system executes and is rendered in the hardware section. These diagrams are typically used to show how components are configured at runtime.

5. Package Diagram

This diagram is a particular type of class diagram. Package diagrams use the same notation as class diagrams, but this diagram focuses on how classes and interfaces are grouped together.

6. Object Diagram

Object diagrams use the same syntax as class diagrams and show how actual instances of a class relate to specific examples of time.

2.2 Unified Process

A unified Process is a way to recognize the importance of communication with customers (customers) and shorten the description of the system from the customer's point of view. The Unified Process emphasizes the important role of software architecture and helps designers focus on goals such as understandability, trust in future changes, and reuse [22], [23].

2.3 Medical Needs Implementation Approach

The approach to looking at the needs of the medical field to be implemented in a web application can be divided into two approaches, namely the integration-convergence approach and the independent approach [24], [25]. In the following, the two methods will be described in more detail.

This approach ensures consistent transfer and integration of information through the organization's overall structure without causing obstacles or limitations in individual units and thereby enabling the achievement of the three main objectives of a leading health information system. First, from an organizational point of view, the purpose is to provide the necessary support to individual units. Second, from an operations point of view, the goal is to facilitate users in carrying out their activities by reducing the work of "servants" and emphasizing the professional aspects of various roles. This will also simplify the introduction to the system in an organizational context and reduce the need for training. Lastly, from a technology point of view, the purpose is to enable the construction of an entire distributed, modular, and evolutionary framework based on open criteria and the use of the adequate architecture, information and technology solutions emerging in the international scenario.

2.4 Developed system

This section contains a problem analysis conducted on the theoretical basis described in the previous chapter. This chapter aims to find requirements in the medical field that can be implemented into web applications and can be applied to all medical institutions whose results will be used to design software that will be implemented in research work.

2.5 Medical World Challenge

When the world of web applications has not hit the medical field, the needs demanded by the medical world are only medical record data. The market continues to grow along with the rapid development of internet technology and web applications. The new requirement, for example, is that patient identification and medical-related records are available in many information systems and infrastructures. At the same time, the confidentiality of information, unique patient identification, and on-demand access to appropriate information is the foundation for medical applications. When web applications began to develop, web applications began to be used, including in the medical field. As medical needs expand, medical web applications must also expand and adapt to meet growing demands.

A. Analysis of Medical Needs Implementation Approach

The theoretical basis explanations regarding the two approaches to implementing medical needs are analyzed, and the results are for implementation. Research and the current reality is that the independent approach is more widely used, and very few are still using the integration-convergence method because the integration-convergence process has one very crucial weak point factor, namely complexity. The development

of a system that becomes the integrator and converter for the web application must be able to be a bridge for other applications and modules made by different suppliers, which will lead to longer development times and problems regarding system independence.

B. Universal Medical Needs Analysis

Each medical institution has different needs in the medical field, depending on the size of the business process, the large or small number of employees, the complex or straightforward system used, and the management method applied to the agency. The impact is that all medical institutions can use not all needs, so each need is individual to the medical institution that uses it. However, it turns out that not all of these medical needs are individual. This means that there are needs that are owned and used by several medical institutions or even needs that are shared by all of them.

Table 1. Comparison of Medical Needs Between Medical Agencies

Module	RS	Clinic	Public health center	Pharmacy
Personnel	S	B	B	J
Online Medical Records Archive	S	B	B	T
Inpatient Management System	S	J	J	T
Outpatient Management System	S	B	B	T
Pharmacy	B	H	H	S
Laboratory	B	H	J	J
radiology	H	H	J	T
User	S	S	S	S

Table 1 has the following information: S = always exists, B = usually exists, H = only specifically exists, J = rarely or slightly exists, and T = none.

2.6 System General Description

The software developed in this research is a web application in the form of a CMS based on JSP. This software is deployed on a computer that becomes a server. The computer that becomes the server must have a special JSP web server and database management system as a supporting application for this software. The server computer must also be connected to a network connection, at least an internal network or Local Area Network (LAN), so that client computers can access this software through a web browser.

The main purpose of developing this software is to meet the universal medical needs needed by all medical institutions, namely to help and facilitate medical activities that can be implemented into web applications so that they become online, more accessible, safer, faster, more efficient, and more reliable. Also, with the advantages of this software, namely the ease of maintenance and customizable, it is hoped that later it will indirectly improve the quality of service and medical treatment from medical institutions that use this software.

2.7 Software Development Analysis

A. Analysis of Software Key Features

The software to be developed must have the main features that are universally applicable to all medical institutions, namely software user management, agency personnel data management, patient outpatient data management, and patient medical record data archiving, which is added with features for additional module management. Meanwhile, the features that will not be implemented are patient inpatient management, pharmacy data management, laboratory management, and radiology management.

B. Software Functional Requirements Analysis

So that software features can be met and software users can use the software properly, the software developed must have several requirements that must be available in the software, which is called the software requirement specification (SRS). SRS is divided into two, namely functional SRS and non-functional SRS. Operational SRS is a task that software must do, while non-functional SRS is a standard or quality that software must have. The available software SRS in this study, from now on referred to as U-Medical, is based on the results of the analysis in the previous subsection and has been mentioned in Table 2, namely:

Table 2. Functional Needs

SRS ID	SRS name	Description
UM-F-01	Personal data change	Users who use this system can change their data
UM-F-02	User data management	Admin can perform user data management in medical institutions that use this system
UM-F-03	Employee data management	Admin can manage employee data in medical institutions that use this system
UM-F-04	Patient data management	Non-admins can manage patient data in medical institutions that use this system
UM-F-05	Diagnostic type data management	Non-admins can manage diagnosis types in medical institutions that use this system
UM-F-06	Patient diagnosis data management	Non-admins can manage patient diagnosis in medical institutions that use this system
UM-F-07	Additional module management	Admin can perform additional module management in this system

while the non-functional SRS of U-Medical software is in table 3.

Table 3 Non-functional Requirements

SRS ID	SRS name	Description
UM-NF-01	*User-friendly*	The system has a good look and interface and can be customized so that it is easy to use
UM-NF-02	Data validation	The system is able to validate user input data based on system specifications
UM-NF-03	Security	The system has user authentication mechanisms, user module access rights management, and delivery and database security
UM-NF-04	JSP-Ajax	The system uses the latest JSP and Ajax technologies which are able to provide a fast response to user actions (less than 5 seconds) and make it easy to build this system

3. RESULTS AND DISCUSSION

This section discusses the implementation of the software based on the analysis and design done in the previous chapter and the testing of the software. The discussion on software implementation includes the implementation environment, implementation limitations, class implementation, and interface implementation, while the panel on software testing includes testing methods, test environments, test objectives, test cases, test results, and analysis of test results.

3.1 Interface Design

The Front Page Interface Design is shown in Figure 1.

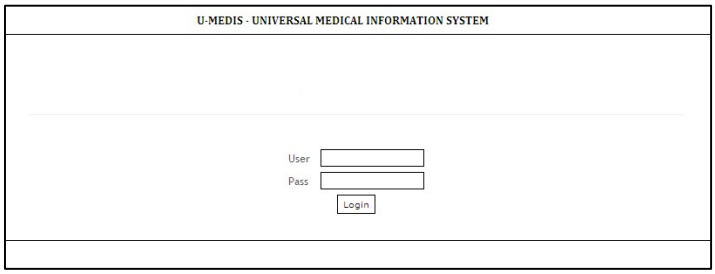

Figure 1. Front Page Interface Design

The Admin Panel Interface Design is shown in Figure 2.

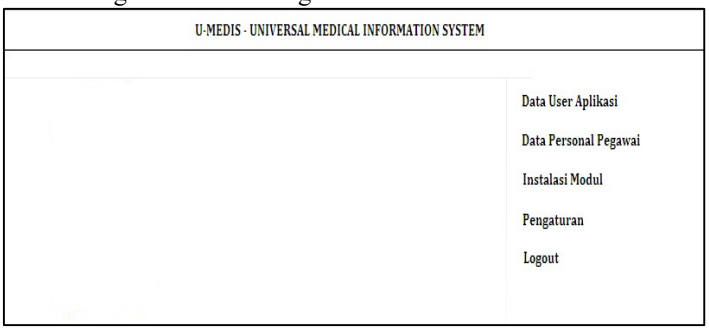

Figure 2. Admin Panel Interface Design

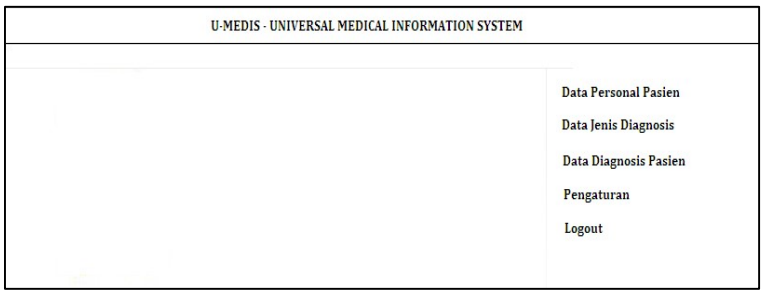

Figure 3. Non-Admin Panel Interface Design

3.2 Software Implementation

A. Implementation Environment

This section discusses and analyzes the software and hardware requirements of the developed system. The software environment used in implementing U-Medical software has the following specifications:

1. Operating system Microsoft Windows XP Professional Edition Service Pack 3 / Ubuntu 9.10 kernel 2.6.31-20
2. Java Standard Edition Software Development Kit 6.0.17
3. Netbeans 6.8 development tool
4. Apache Tomcat Server 6.0.20
5. XAMPP 1.7.2
6. MySQL Database 5.1.37
7. Mozilla Firefox browser 3.5.8

While the hardware environment specifications used in the development of U-Medical software are as follows:

1. Processor Intel Core 2 Duo E4600 @2.40GHz
2. Memory DDR2 RAM 1 GB
3. 80GB IDE Hard Drive
4. The output device is a monitor
5. Input devices such as mouse and keyboard
6. The minimum software requirements needed to run U-Medical software are as follows:
7. Microsoft Windows XP Home Edition operating system, Linux kernel variant 2.6.23, or Mac OS X
8. The operating system has been installed Java Runtime Environment 6.0.10, Apache Tomcat 6.0.18, and MySQL 5.1.20
9. W3C standard browser

While the minimum hardware requirements needed to run the U-Medical software are as follows:

1. Intel Pentium M or AMD Athlon Processor or equivalent
2. Memory DDR RAM 512MB
3. Free hard disk storage 3 MB
4. The output device is a monitor
5. Input devices such as mouse and keyboard
6. Implementation Limits

The constraints defined for the implementation of U-Medical software are as follows:

1. The connection made by the web server is only through the local or internal network (LAN).
2. The type of file that is uploaded must match the category that has been defined at the place where the file is uploaded.
3. The addition of the module is still by uploading the files owned by the module separately according to the file type.

B. Class Implementation

The described class design is then implemented in a file with the java extension. These classes are very closely related to one another. Therefore, changes to the software must be made directly to the source code and recompiled (hard-coded). Class mapping can be seen in Table 4.

Table 4 Class Implementation

Class name	File Name
panel	panel.java
dbconnect	dbconnect.java
admin	admin.java
diagnosis	diagnosis.java
foto	foto.java
kontrol	kontrol.java
lab	lab.java
modul	modul.java
pasien	pasien.java
pegawai	pegawai.java
silang	silang.java
terapi	terapi.java

C. Interface Implementation

The implementation of the U-Medical software interface is carried out in accordance with the described software design by taking into account aesthetic and usability factors to support software functionality. Figure 4 shows the Implementation of the Front Page Interface.

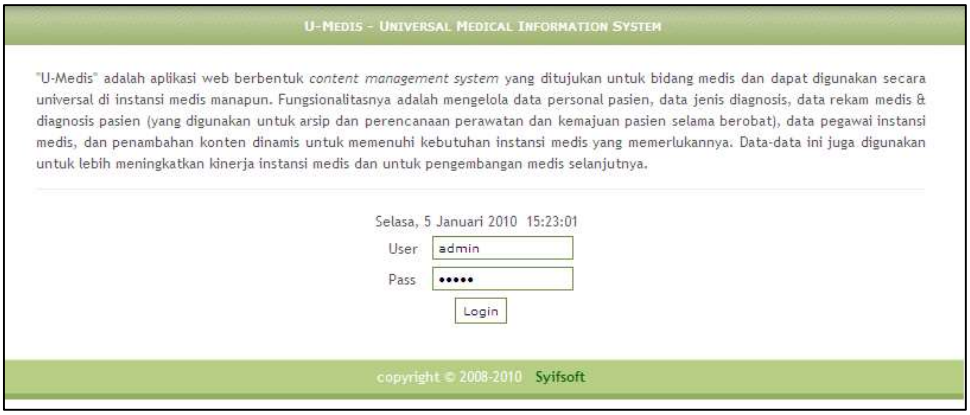

Figure 4. Implementation of the Front Page Interface

Figure 5 shows the Implementation of the Admin Panel Interface.

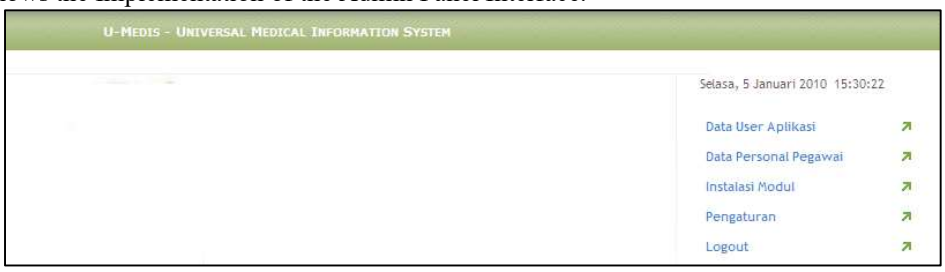

Figure 5. Admin Panel Interface Implementation

Implementation of the Non-Admin Panel Interface is shown in Figure 6.

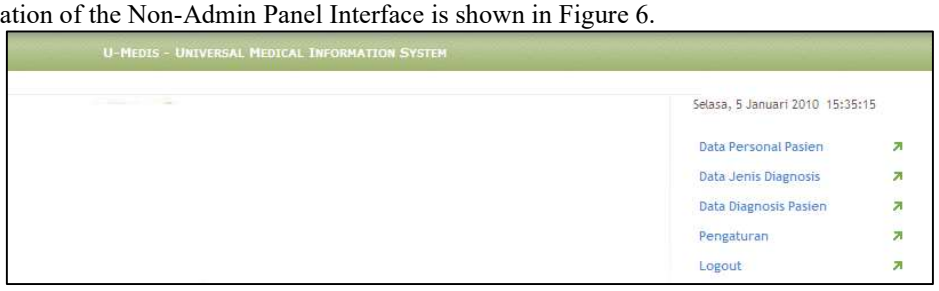

Figure 6. Implementation of the Non-Admin Panel Interface

Implementation of the Application User List Interface is shown in Figure 7.

Figure 7. Implementation of the Application User List Interface

Implementation of the Add User Interface Application is shown in Figure 8.

Figure 8. Implementation of the Add User Interface Application

The implementation of the Diagnostic Image Change Interface is shown in Figure 9.

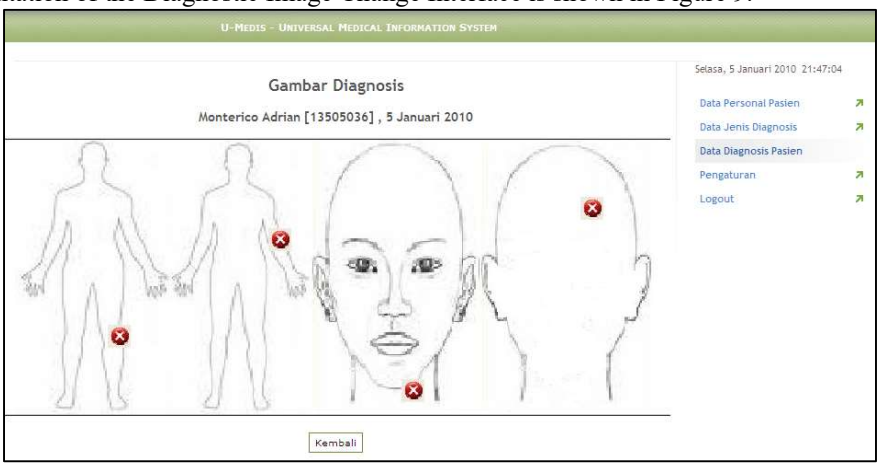

Figure 9. Interface Implementation Change Image Diagnosis

3.3 Software Testing

This section contains the testing methods used in this study.

A. Testing Method

The testing method used is black-box, and there is no difference in the way of testing with similar CMS web applications or CMS web applications that are not specifically for the medical field, namely by checking the suitability of the results of the software implementation with the stated requirements specifications.

B. Testing Environment

The U-Medical software testing environment has the exact specifications as the software implementation environment described.

C. Testing Purpose

The purpose of testing the U-Medical software developed in this research is to check whether the software implementation results are in accordance with the requirements specifications mentioned.

D. Test Case

Testing of U-Medical software must meet the first test objective. Namely, software functionality testing is carried out to check the suitability of the U-Medical software implemented with the results of software analysis and design. This test is carried out to ensure that the U-Medical software has met all the functional

requirements that have been defined. Tests are carried out according to the scenarios for each previously described use case.

E. Test result

The test results of each use case can be seen in the following table, and the details of the identification of the test cases.

Table 5. Identification of Test Cases

Use Case	Test Case Identification	Results
UM-UC-01	U-UC-01-01	Accepted because the results obtained by all test cases are in accordance with the expected results
UM-UC-02	U-UC-02-01, U-UC-02-02	Accepted because the results obtained by all test cases are in accordance with the expected results

The results obtained from the functionality testing that has been done follow the software's defined functional requirements. The entire process that supports the software's functionality can be appropriately executed and produces output that follows the expected results.

F. Use Case Changing personal data

*Use Case*UM-UC-01 has User as Actor with Pre-condition User has logged into the system.

Table 6. Use Case Scenarios Changing personal data

Actor Action	System Reaction
Normal Scenario (S-UC-01-01)	
1) Click the Settings menu	
	2) Displays a form that can be filled in with the user's personal data
3) Enter the old password, fill in the fields you want to change, press the Save button	
	4) Validate old password and filled fields, save new personal data to database, and display success message
Alternative Scenario (S-UC-01-02)	
	4a) Displays an error message because the old password is wrong or the data entered is invalid

G. Use Case Manage user data

*Use Case*UM-UC-02 uses Admin as Actor with Pre-condition Admin has logged into the system.

Table 7 Use Case Scenarios Managing user data

Actor Action	System Reaction
Normal Scenario (S-UC-02-01)	
1) Click the Application User Data menu	
	2) Displays a list of users as well as buttons to add or delete user data
3) Click the Add User button	
	4) Displays a form that can be filled in with the new user's personal data
5) Fill in the new user's personal data and press the Save button	
	6) Validate data on form and save new user data to database
Alternative Scenario (S-UC-02-02)	
	6a) Displays an error message because the username already exists or the data entered is invalid
Alternative Scenario (S-UC-02-03)	
3b) Click the Delete button on the user you want to delete	
	4b) Displays a message box for confirmation
5b) Click OK button	
	6b) Selected user data is deleted from the database

4. CONCLUSION

Things that can be concluded during the work of this research are as follows. Medical needs that can be implemented into a web application and can be universally applicable to all medical institutions are the needs of medical agency personnel data management, online medical record archives, outpatient data management, application user data management, and additional application management modules. For research

development, suggestions that can be given are as follows. The "U-Medical" software can be further developed into a universal web application if there is further reference to new universal medical needs.

ACKNOWLEDGEMENTS

The authors would like to thank the Bandung Institute of Technology and Telkom University.

REFERENCES

[1] A. Milenkovic, D. Jankovic, and P. Rajkovic, "Extensions and adaptations of existing medical information system in order to reduce social contacts during COVID-19 pandemic," *Int. J. Med. Inform.*, 2020, doi: 10.1016/j.ijmedinf.2020.104224.

[2] Contentmanager, "Enterprise Content Management," 2008. .

[3] S. Gong, X. Zhu, R. Zhang, H. Zhao, and C. Guo, "An Intelligent Resource Management Solution for Hospital Information System Based on Cloud Computing Platform," *IEEE Trans. Reliab.*, 2022.

[4] J. Dey, A. Sarkar, S. Karforma, and B. Chowdhury, "Metaheuristic secured transmission in Telecare Medical Information System (TMIS) in the face of post-COVID-19," *J. Ambient Intell. Humaniz. Comput.*, 2021, doi: 10.1007/s12652-021-03531-z.

[5] B. Liu, M. Mohandes, H. Nuha, M. Deriche, F. Fekri, and J. H. McClellan, "A Multitone Model-Based Seismic Data Compression," *IEEE Trans. Syst. Man, Cybern. Syst.*, 2021, doi: 10.1109/tsmc.2021.3077490.

[6] S. Pronin *et al.*, "Does a history of lumbar spine surgery predict radiological cauda equina compression in patients undergoing MRI for suspected cauda equina syndrome?," *Br. J. Neurosurg.*, 2020, doi: 10.1080/02688697.2019.1687845.

[7] Bo Liu, Mohamed Mohandes, and Hilal Nuha, "System, method and computer program product for compression of seismic data," US Patent App. 16/670,477, 2021.

[8] A. Riaz *et al.*, "ExpressionHash: Securing Telecare Medical Information Systems Using BioHashing," *Comput. Mater. Contin.*, 2021, doi: 10.32604/cmc.2021.014418.

[9] F. E. Ahmed, R. Hashaikeh, and N. Hilal, "Solar powered desalination – Technology, energy and future outlook," *Desalination*. 2019, doi: 10.1016/j.desal.2018.12.002.

[10] A. Coronato and A. Cuzzocrea, "An Innovative Risk Assessment Methodology for Medical Information Systems," *IEEE Trans. Knowl. Data Eng.*, 2022, doi: 10.1109/TKDE.2020.3023553.

[11] R. Amin, S. K. H. Islam, P. Gope, K. K. R. Choo, and N. Tapas, "Anonymity Preserving and Lightweight Multimedical Server Authentication Protocol for Telecare Medical Information System," *IEEE J. Biomed. Heal. Informatics*, 2019, doi: 10.1109/JBHI.2018.2870319.

[12] K. A. Bhavsar, J. Singla, Y. D. Al-Otaibi, O. Y. Song, Y. Bin Zikria, and A. K. Bashir, "Medical diagnosis using machine learning: a statistical review," *Computers, Materials and Continua*. 2021, doi: 10.32604/cmc.2021.014604.

[13] H. F. Hasya, H. H. Nuha, and M. Abdurohman, "Real Time-based Skin Cancer Detection System using Convolutional Neural Network and YOLO," 2021, doi: 10.1109/IC2IE53219.2021.9649224.

[14] M. Mohandes, S. Rehman, H. Nuha, M. S. Islam, and F. H. Schulze, "Accuracy of Wind Speed Predictability with Heights using Recurrent Neural Networks," *FME Trans.*, 2021, doi: 10.5937/FME2104908M.

[15] S. M. Mousavi and G. C. Beroza, "A Machine-Learning Approach for Earthquake Magnitude Estimation," *Geophys. Res. Lett.*, 2020, doi: 10.1029/2019GL085976.

[16] T. C. Hsiao, Z. Y. Wu, Y. F. Chung, T. S. Chen, and G. B. Horng, "A secure integrated medical information system," *J. Med. Syst.*, 2012, doi: 10.1007/s10916-011-9793-z.

[17] WHATWG, "HTML5 Draft Standard." http://www.whatwg.org/specs/web-apps/current-work (accessed Oct. 26, 2009).

[18] N. F. Malik, A. Nadeem, and M. A. Sindhu, "Achieving State Space Reduction in Generated Ajax Web Application State Machine," *Intell. Autom. Soft Comput.*, 2022, doi: 10.32604/iasc.2022.023423.

[19] E. Zivkovic, "Content Management Systems Overview." www.ezau.com/latest/articles/0235.shtml (accessed Oct. 26, 2009).

[20] B. D. Pilone and N. Pitman, *UML 2.0 in a Nutshell*. 2005.

[21] M. H. L. Vo and Q. Hoang, "Transformation of uml class diagram into owl ontology," *J. Inf. Telecommun.*, 2020, doi: 10.1080/24751839.2019.1686681.

[22] I. Jacobson, G. Booch, and J. Rumbaugh, *The Unified Software Development Process*. Addison-Wesley, 1999.

[23] S. Ain El Hayat, F. Toufik, and M. Bahaj, "UML/OCL based design and the transition towards temporal object relational database with bitemporal data," *Journal of King Saud University - Computer and Information Sciences*. 2020, doi: 10.1016/j.jksuci.2019.08.012.

[24] F. Fedele, "Healthcare and Distributed Systems Technology," 1995. http://www.mclink.it/hltcare.pdf (accessed Feb. 04, 2009).

[25] R. Wei, A. Zolotas, H. Hoyos Rodriguez, S. Gerasimou, D. S. Kolovos, and R. F. Paige, "Automatic generation of UML profile graphical editors for Papyrus," *Softw. Syst. Model.*, 2020, doi: 10.1007/s10270-020-00813-6.