ISSN 2355-0082

News Management Application Development

Muhammad Adryan Syahputra[1], Rizqa Raaiqa Bintana[2*], Dewi Lestari[3]
[1,2,3] Information Systems Study Program, Universitas Jambi, Jambi, Indonesia
[*]rizqa.raaiqa.bintana@unja.ac.id

Accepted 11 November 2024
Approved 14 November 2025

***Abstract*— This study aims to develop a news management application at institution X that has been running previously, into an interactive application that can be used as a means of writing, reviewing and collecting news articles which are then published to daily newspaper pages. The application development model used in this study is the waterfall model. The waterfall model involves stages such as communication, planning, modeling, construction, and deployment. This study succeeded in developing a news management application from the previous version into a new application with the implementation of all its functional requirements. Testing was carried out using the black box method and system usability scale (SUS). The results of the black box test showed that the news management application had successfully carried out all functions correctly and in accordance with expectations. In the SUS test, two measurements were taken, namely before and after development. The SUS score before development was 71.3, while after development it increased to 78.8. This shows an increase in user perception of the usability of the application after development was carried out. Structured work and a clear flow from initiation to end can minimize errors because it has clear details and final descriptions.**

***Index Terms*— black box testing; news application; system usability scale; waterfall model**

I. INTRODUCTION

Institution X seeks to improve public services, especially in terms of providing various information related to statistical developments in various sectors in one area where institution X is located. Information is presented regularly and transparently so that the public can easily find out or obtain the data or information needed. Information is distributed from institution X to the public through the daily news page in the area where institution X is located. News articles that will be published on the daily news page are written directly by employees at institution X. Before being published, the news articles must go through a review stage, where the news articles will be reviewed by reviewers from the institution itself.

Previously, institution X had an application that could be used to collect and manage news articles from employees who had been assigned until the review process before the news articles were published on the daily news page. The news management application is website-based. The application has a primary role as an application that is expected to facilitate the process of managing news articles written directly by employees. News articles are created with the help of third-party applications such as Microsoft Word and the like, which are then uploaded to the application in the form of files with the doc and docx extensions. The news writing can be managed by the reviewer by downloading the file first. Then, the reviewer reviews the content of the news, the completeness of the news writing, until the news writing can be said to be ready to be published by the daily newspaper.

This news management application aims to meet several needs. One such need is the ability to write news directly within the application without having to use a third-party application like Microsoft Word. Therefore, the purpose of this development is to refine the process so that all news management activities can be done only within a single news management application, including writing news text, editing news files and reviewing news articles without the need to upload and download documents. Development here means activities to create a new system or application that can replace the old system as a whole or improve the existing system [1].

Application development can use various forms of models, one of which is using the waterfall model. The waterfall development model is a model that provides a sequential or ordered software life flow approach starting from analysis, design, coding, and testing like a waterfall [2]. The waterfall model is suitable for use in projects that have clear and well-defined requirements and needs. This model is also easy to organize tasks and processes and development results are well documented [3]. This waterfall development model has been used in previous studies, one of which was conducted by Rifai & Mailasari [4] regarding the use of the waterfall method in the sales and purchase data processing information system. This study aims to create a system that can facilitate sales activities and data collection of goods at the Fang Fifi Cell cellphone store by adapting the waterfall model in its development to ensure that the development process was systematic and organized.

A similar study using the waterfall model was conducted by Septyanto et al [5], who aimed to develop a website-based e-learning system and determine the feasibility of the resulting product. Purwati et al [6] successfully designed a website-based news

information system by regenerating the layout of an old news website with a new design, resulting in a more modern layout and ease of use.

The development of a news management application for institution X has clear needs, namely the need for several features such as news writing features and several supporting features for news management that are not available in the previous application. Therefore, the selection of the waterfall model is suitable for the development of this application because the waterfall model has a fixed (definite) development process stage, is easy to apply, is orderly and is suitable for software products or programs that have clear requirements from the start so that errors are minimal [7].

TABLE I. COMPARISON OF SOFTWARE DEVELOPMENT MODELS

Waterfall	Prototype	Rapid Application Development	Scrum
Sequential	Iterative	Iterative	Iterative
For software projects with clear requirements from the outset	For customized software	For software projects that require a short time	For various sizes and types of software projects and those requiring short turnaround times
The development process stages are fixed, easy to apply and the process is regular	Shorten software product development time	Can reuse existing components	Working with time constraints called sprints
Cannot proceed to the next stage if a stage has not been completed	Provide a prototype as an illustration of the system to be built	Only compatible with modularizable software	Focus on team spirit and communication
It is difficult to experience changes in the needs desired by clients	Changes can be made while the system or software is still in prototype	If there are changes in the middle of the work, a new contract must be made between the developer and the client	Hold frequent meetings to get feedback from clients and stakeholders

Table I shows a comparison of software development models summarized from previous research [7][8]. Based on the characteristics and comparison of several development models described in Table I, the waterfall model was chosen for this application's development. This model was chosen because the goals and needs of the news management application were clearly defined at the beginning of development. The waterfall model also has fixed stages, is easy to apply, and has a structured process. Furthermore, the application being developed is relatively small.

Previous applications will be measured for their usability level using the system usability scale (SUS). Then, the application that has been developed will be tested using two types of testing methods, namely black box and SUS. In black box testing, the focus is on checking whether the application functions as expected. The tester tests the application by testing predetermined input conditions to verify whether the application behaves according to its functional specifications [9]. Meanwhile, the system usability scale is a questionnaire that can be used to measure the usability of a computer system from the user's subjective perspective [10]. The results of the SUS test on the previous application will be compared with the results on the latest application to determine whether there is an increase in usability after the application is updated. After being tested, the application that has been developed is then implemented on the server belonging to institution X and receives feedback from users on the application that has been developed. With the development of this application, it is hoped that it can become a website-based application that can be used to facilitate the process of writing, managing and reviewing news articles at institution X. This research adopted the waterfall model in its development. All the stages of the waterfall model process will be presented in detail throughout this article, helping readers understand the details of the stages within the waterfall model.

II. METHOD

The research framework begins with identifying the problem, then conducting a study literature and developing a news management application using the waterfall model. The research framework can be seen in Figure 1.

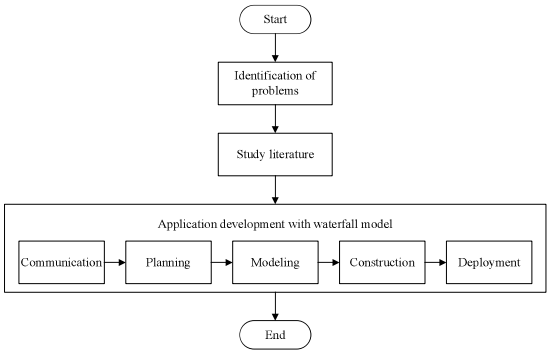

Fig. 1. Research framework

A. Communication

At the communication stage, a discussion was held with institution X to understand the objectives and

needs of the development of the news management application. This discussion was conducted directly, where this discussion resulted in a description of the flow and features of the previous news management application and what shortcomings needed to be fixed in this development.

B. Planning

At the planning stage, planning is carried out to develop a news management application by considering the time and resources available. Discussions are carried out with relevant stakeholders to determine the estimated time needed to develop the application. Furthermore, a development schedule is made. The entire planning process is carried out in a structured and systematic manner to ensure the success of the development of the news management application.

C. Modeling

Requirement analysis is a very important stage in software development. If there is an error in describing the system requirements, it can result in the failure of software development [11]. Requirements are classified as functional requirements and non-functional requirements [12]. At the modeling stage, analysis and design of news management applications are carried out. First, a requirements analysis is carried out to determine the functional and non-functional needs of the news management application to be updated.

After analyzing the required needs, the next step is to design the news management application. This stage is more focused on the design of the updated application, which includes the database structure, the updated appearance of the application, and the updated application workflow. The design will be described in the form of UML and wireframe.

Use case diagrams are used to determine the behavior of the system to be implemented. Use case diagrams consist of actors, use cases, and their relationships. Diagrams are used to model the system/subsystem of an application. Use case diagrams fulfill certain functionalities of a system [13]. Classes represent the fundamental building blocks of any object-oriented software system. UML class diagrams offer a clear and effective means of illustrating these components. In addition to detailing individual classes, class diagrams demonstrate the relationships among multiple classes, thereby depicting the overall architecture of the system. A class is characterized by a unique name, a set of attributes, and a list of methods [14].

Activity diagrams are valuable behavioral diagrams that illustrate the internal processes of various program operations using nodes and edges. These diagrams have been widely utilized across different domains to represent workflows effectively. An activity diagram serves to depict various activities, sub-activities, transitions, decision points, guard conditions, concurrent activities, branches, merges, swim lanes, joins, forks, and more. Typically, an activity diagram begins with a single starting activity and culminates in a single final activity [15]. The new design produced will still adopt several elements of the old design that are considered still suitable for use.

D. Construction

At the construction stage, program code and testing are carried out on the development of news management applications. The creation of program codes is adjusted to the results of the requirements analysis and design that have been carried out in the previous stages, so that the input and output produced in the development of website-based news management applications can be implemented using web programming properly. The programming language and type of database will be adjusted to the type that has been implemented in the application that has been running previously.

The news management application that has been developed will be tested in terms of functionality and the level of usability of the application from the user's perspective. The application's functionality is tested using the black box testing method, while the level of usability of the application is tested using the system usability scale (SUS) method.

- Black Box Testing. The purpose of conducting black box testing on the news management application is to find out whether each function works well by determining the test case data being tested. Furthermore, determining the test scenario used to evaluate the output produced in the news management application. This testing is documented in the form of a table with the format in Table II [16]. The black box questionnaire will be distributed to respondents from institution X representing each role in the news management application, including 1 respondent as a news writer, 1 respondent as a news reviewer, 1 respondent as a publisher and 1 respondent as an admin.

TABLE II. BLACK BOX TESTING FORMAT

No	Test Scenario	Test Case	Expected Results	Test Results
1	The scenario that is executed	Specifications of the data used	Expected test scenario results	As expected or not

- System Usability Scale (SUS). The level of usability of the news management application was tested by providing 10 statements that had to be answered by respondents. These statements were adopted based on research conducted by Bangor [17]. The SUS questionnaire will be distributed to 18

respondents from institution X who had previously used the previous version of the news management application. The SUS test was conducted twice, the first for the previous news management application and the second for the application that had been developed, to compare the level of usability in each application. The System Usability Scale is a usability evaluation method that can provide adequate results based on considerations of small sample sizes, time and costs [18]. In calculating the SUS score, a five-point Likert scale (1-5) is used, consisting of Strongly Disagree, Disagree, Neutral, Agree, and Strongly Agree. Each statement item has a contribution score ranging from 0 to 4. For items 1, 3, 5, 7 and 9 (positive statement items), the contribution score is the scale position minus 1. For items 2, 4, 6, 8 and 10 (negative statement items), the contribution score is 5 minus the scale position. Then multiply the total SUS score by 2.5 to obtain the overall SUS score, which ranges from 0 to 100 [10]. The following are the steps taken in converting respondent responses [19]:

- Odd statements, namely: 1, 3, 5, 7 and 9, the score given by the respondent is minus by 1.

  Odd SUS score = ∑Px-1, where Px is the x-th odd statement scale.

- Even statements, namely: 2, 4, 6, 8 and 10, the scores given by respondents are used to minus 5.

  Even SUS score = ∑5-Pn, where Pn is the n-th even statement scale.

- The results of the conversion are then added up for each respondent and then multiplied by 2.5 to obtain a value range of 0-100.

  (Odd SUS Score + Even SUS Score) x 2.5

- Once each respondent's score is known, the next step is to find the average score by adding all the respondents' scores and dividing by the total number of respondents. This calculation can be seen in (1).

$$X = \sum x / N \quad (1)$$

Description:
X : Average score
∑x : Total system usability scale scores
N : Number of respondents

E. Deployment

At this stage, the news management application that has been developed is then implemented to the server owned by institution X and data migration occurs as the news management application is used. Migration occurs on news data that was originally in the form of files, then converted into text that can be stored in the database.

TABLE III. SUS SCORE ASSESSMENT AND EXPLANATION

Acceptability Ranges		
Score	**Term**	**Meaning**
70-100	ACCEPTABLE	Shows good levels of acceptance and usability, although there is still improvement
63-69	MARGINAL HIGH	Although there are still shortcomings, the product can still be improved and is considered to have a limited level of usefulness
50-62	MARGINAL LOW	The product has many serious flaws that need to be fixed immediately and is considered to have a low level of usability
0-49	NOT ACCEPTABLE	Products with this score have many weaknesses in use and need significant improvement
Grade Scale		
90-100	A	Very good
80-89	B	Good
70-79	C	Sufficient
60-69	D	Lacking
0-59	F	Very lacking
Adjective Ratings		
85.58 - 100	BEST IMAGINABLE	The best quality imaginable
72.76 - 85.57	EXCELENT	Very good quality
52.02 - 72.75	GOOD	Good quality
39.18 - 52.01	OK	Quite good quality
25.01 - 39.17	POOR	Poor quality
0 - 25.00	WORST IMAGINABLE	The worst quality imaginable

Reference: Bangor et al. [17]

III. RESULTS AND DISCUSSION

A. Communication

In the early stages of developing a news management application, a project initiation was carried out with related parties, namely institution X as the owner of the application. Communication was carried out to find several weaknesses in the old application, such as the lack of features needed by users. These things make the old application inefficient and do not meet the needs of managing news writing. Therefore, it is necessary to develop a news management application that aims to overcome these weaknesses and provide more complete features.

The previous news management application used by institution X consisted of four main roles, namely news writers, news reviewers, and news publishers along with the admin role. News writers are tasked with writing news and uploading it in the form of doc or docx files into the news management application. News

reviewers are tasked with reviewing news articles that have been uploaded by news writers before being published. Meanwhile, news publishers are tasked with selecting news articles to be published, where the news articles have previously been reviewed by news reviewers. The publisher role is given to the daily newspaper. Each role has features that they can run as listed in Table IV.

TABLE IV. ROLES AND FEATURES IN PREVIOUS APPLICATIONS

Features	Role			
	Writer	Reviewer	Publisher	Admin
Add news	✓			✓
Edit news	✓			✓
Reject news		✓	✓	✓
Upload review results		✓		✓
Cancel news review		✓		✓
Download news files and photos		✓	✓	✓
Add and edit published news links			✓	✓
News monitoring	✓	✓	✓	✓
Download monitoring results		✓		✓
Search for news	✓	✓	✓	✓
News details		✓	✓	✓
Login and logout	✓	✓	✓	✓
Edit account profile	✓	✓	✓	✓
Manage system				✓

In previous news management applications, writers uploaded news articles in doc or docx file format by filling in input such as title, photo (optional), news file, author's name, and work unit as supporting data. News articles entered the application with the status "Ready to Review". The reviewer selected the news to be reviewed and downloaded the news file because the content of the news could not be displayed by the application. After conducting the review, the reviewer uploaded the review result file. The reviewed news articles were saved with the status "Ready to Publish". The publisher selected the news articles that were ready to be published and published them on the daily newspaper page. After being published, the publisher entered the news link so that the news status became "Published".

There are several weaknesses found in the previous application. The weaknesses found are as follows:

- The news writing process does not occur in the news management application, causing the writing process to be inefficient and time-consuming.
- The news review process cannot be done in the application. The process is still carried out manually by reviewers outside the news management application.
- There is no feedback or comments given when a news article is rejected, so that news writers do not know what mistakes are made in their news articles.
- There is no notification and comment feature that can notify the status of the news article, so that writers do not know that their news articles have been rejected along with the reasons.
- The application does not display statistical data on news management that is appropriate for each role.
- News is still uploaded in file form so that the contents of the news article cannot be displayed in the news management application.

B. Planning

In the planning stage, planning and scheduling are carried out before the development stage begins. In this stage, information is collected and analyzed regarding the objectives, scope, and estimated time for application development. Application developers collaborate with institution X as the application owner to ensure that business requirements are met and time targets can be achieved.

The purpose of developing news management application is to improve the efficiency and quality of news writing management at institution X. In addition, the development objectives also include the development of new features needed by users, such as news writing features, notifications, and ease of application accessibility. The development objectives also include improving weaknesses in previous applications that have been previously identified. By achieving these objectives, it is hoped that the news management application can provide more optimal benefits for users and application owners.

The scope of development of the news management application includes:

- Redesigning the application workflow.
- Redesigning the appearance that needs improvement.
- Adding new features that make it easier for users to manage news articles, such as writing, commenting and notification features.
- Fixing some program codes that are considered less efficient.

The scope of development is intended to improve the performance and effectiveness of the news management application in managing news articles and providing a better user experience. Table IV is an estimate of the time needed to develop a news management application. After getting an estimate of the time needed to develop a news management application, a structured time management is needed to complete the development on time. Therefore, during the development process, tracking is carried out once a week to monitor progress and evaluate whether there are any delays or obstacles that can hinder development. With this tracking, the development of the news management application can be carried out according to the predetermined schedule and avoid delays in development.

TABLE V. ESTIMATED DEVELOPMENT TIME

Activity	Estimate (Day)
Design:	
Requirements analysis and UML	2
Create a new database design	1
Creating a new interface design	2
Total days	5
Coding:	
Database implementation and coding	14
Testing:	
Black box testing	1
System usability scale	1
Total days	2
Deployment	1
Total application development days	23

C. Modeling

At the modeling stage, the analysis is carried out by producing several functional and non-functional requirements for the development of news management applications. These requirements are the basis for the design stage. The following are the results of the analysis and design of the developing news management application.

- Requirements Analysis

Analysis of news management application development requirement is obtained from the results of observations and evaluations of previous applications. This analysis includes identification of functional and non-functional requirements in the application, as well as improvements needed to overcome weaknesses that have been found in previous applications. The results of this analysis are the basis for designing and developing a new version of the news management application. The functional requirements needed in the development of the news management application are listed in Table VI and the non-functional requirements needed are listed in Table VII.

TABLE VI. FUNCTIONAL REQUIREMENTS

Code	Functional Requirements
KF-01	The application can display complete statistics based on user roles and news writing and review activities on the dashboard
KF-02	The news writing process can be done in the application
KF-03	The application can edit the contents of news articles
KF-04	The application can provide notifications when there is a rejection of a news article
KF-05	The application has a feature to save drafts of news articles
KF-06	The application has a feature to cancel news articles that have been uploaded
KF-07	The news writing review process can be done within the application
KF-08	The application can display news content in detail
KF-09	The application can convert news articles into docx files
KF-10	The application has a comment/feedback feature

TABLE VII. NON-FUNCTIONAL REQUIREMENTS

Code	Non-Functional Requirements
KNF-01	The application has a good appearance and is easy to use
KNF-02	The application uses a responsive web design view
KNF-03	The application must be accessible to users at all times

- Database Design

Database design in the development of news management applications is focused on the mst_berita table only because the mst_berita table is a table used to store news data such as news titles, news authors, news status, and so on. The tables in the news management application database include mst_berita, satker, akses_menu, akses_submenu, akses_user_level, menu, submenu, user, and level. Changes were made to the mst_berita table by adding 3 new fields, namely the isi_berita field which is used to store news content in text form, the komen field is used to store comments given by reviewers and publishers, and the islihat field is used for notification purposes. The design results that occur in the mst_berita table can be seen in Table VIII. Then, Figure 2 shows the relationship between tables in the news management application database.

- Use Case Diagram

Use case diagrams are used to describe the interaction between users and news management applications. Use case diagrams will be divided based on the roles in the news management application and then compared based on the previous application to the application to be developed.

TABLE VIII. NEW MST_BERITA TABLE STRUCTURE

No	Field	Type	Information
1	id	Integer (6)	Unique id of news article
2	user_id	Integer (11)	Id of the user who wrote the news article
3	judul_berita	Varchar (100)	News title

No	Field	Type	Information
4	penulis	Varchar (50)	Author name
5	isi_berita*	Text	The content of the news is stored in HTML text
6	komen*	Text	Comments from publishers and reviewers
7	islihat*	Tinyint (1)	Stores a value if the notification has been seen by the user
8	satker_kd	Varchar (5)	Save the work units of the author
9	tgl_upload	Datetime	Date the article was uploaded by the author
10	status_kd	Integer (2)	Save the status of a news article in the form of numbers
11	tgl_publish	Datetime	Date when the news was published by the publisher
12	link_publish	Varchar (255)	Links to news that has been published
13	editor	Varchar (50)	Name of the reviewer who conducted the review
14	tgl_mulai_review	Datetime	Review start date
15	tgl_selesai_review	Datetime	Review completion date
16	file_draft	Varchar (50)	Save the name of the old news file which is still in file form
17	file_review	Varchar (50)	Save the name of the old news file which is still in file form
18	image_upload	Varchar (200)	Saves the name of the photo uploaded by the user

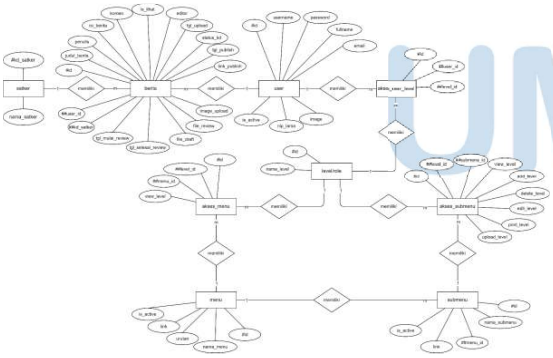

Fig. 2. Database relations

In the use case diagram of the previous news management application, it can be seen that the admin role still has access to all features. Because the news management application has a multirole feature, where one account can have more than one role. Thus, the admin role should not be able to carry out the process of adding news articles, reviewing news articles to publishing news articles. Therefore, the admin role can be focused on taking relevant actions.

After being developed, there are significant differences in the use case diagram between the news writer and admin roles. News writers can write news articles, access news article details, and add news drafts. While the admin is now more focused on the application management function. For the reviewer and publisher roles, their tasks in the previous news management application remain the same as those developed, but differences are seen in the activity diagram because changes occur in the workflow. Figure 3 and Figure 4 are use case diagrams for the news writer and admin roles in the previous news management application. Meanwhile, Figure 5 and Figure 6 are use case diagrams for the roles of news writers and admins in the latest version of the news management application.

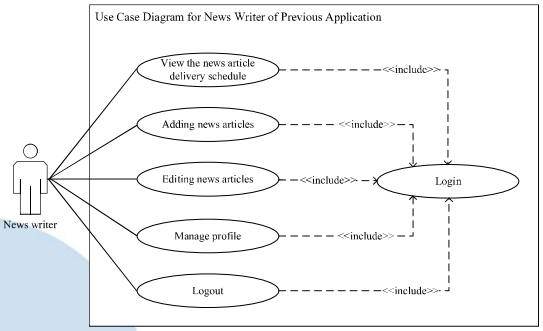

Fig. 3. Use case diagram for news writer of previous application

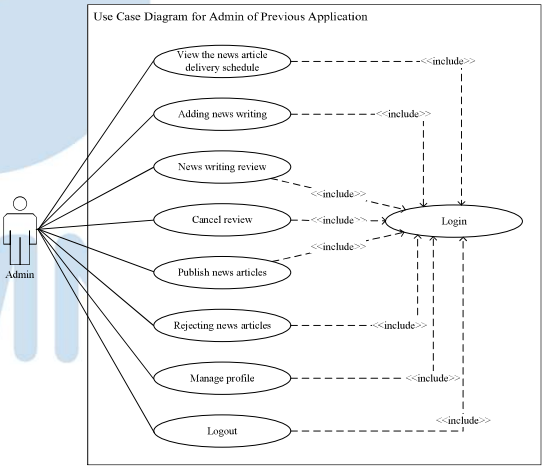

Fig. 4. Use case diagram for admin of previous application

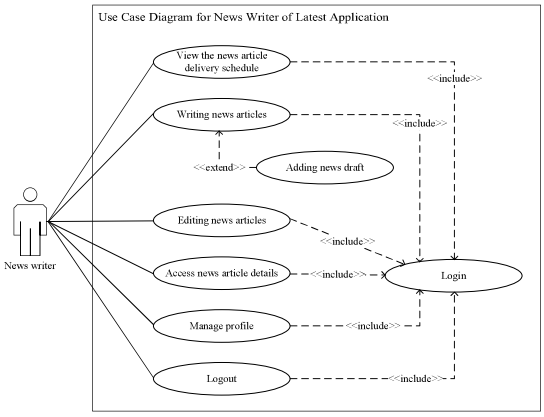

Fig. 5. Use case diagram for news writer of latest application

- Class Diagram

At the design stage, a class diagram is created to describe the class structure used in the application. The design of the class diagram will be focused on the developed part only, namely the masterBerita and masterDashboard sections. The class diagram is also the basis for the program code used in developing a news management application. This class diagram can be seen in Figure 7.

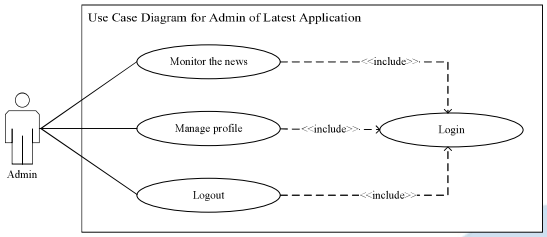

Fig. 6. Use case diagram for admin of latest application

Fig. 7. Class diagram

- Activity Diagram

In the application design stage, activity diagrams are used to describe the process flow that occurs in the application. Activity diagrams show all stages from data input to the final process carried out by the application. In the case of news management development, the activity diagram of the previous application is compared to the latest application. Activity diagrams that have changed in the previous process flow include the process of adding news articles, editing news articles, reviewing news articles, publishing news articles and the process of rejecting news articles. The results of the process changes are shown in Figures 8 to 12. There are new activities such as adding draft news articles as shown in Figures 13 and 14.

- User Interface

In the interface design stage, improvements were applied to parts that were considered less than optimal and needed development in the previous news management application. Pages that did not change during development such as the login page, profile page, author news list page and reviewer news list page.

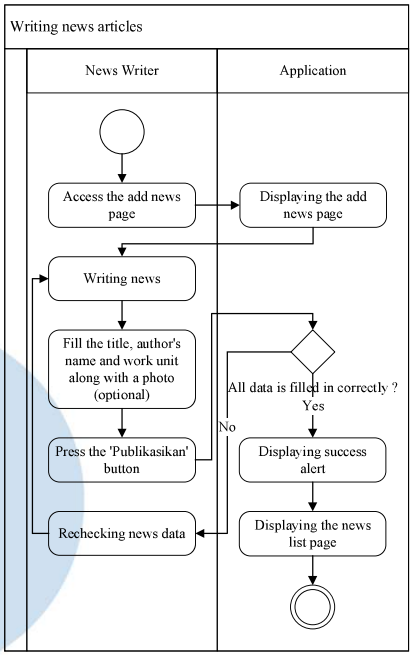

Fig. 8. Activity diagram for writing news of updated process flow

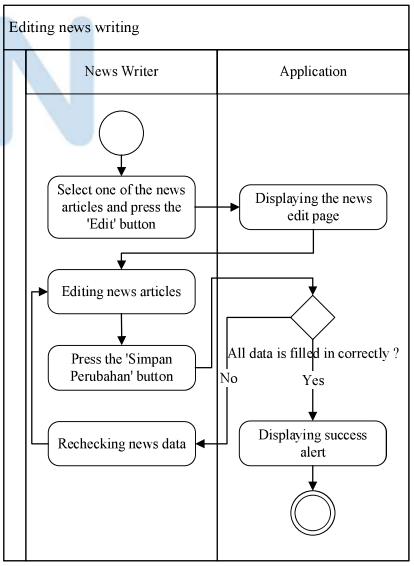

Fig. 9. Activity diagram for editing news of updated process flow

The design of the developed news management application interface focused on rearranging the appearance of pages that needed improvement and designing the appearance of several new features. Some

of the displays that experienced the most significant changes were the appearance of the dashboard page, the add news page, the final review page and the news detail page. Previously, the dashboard page for each role had the same appearance. Now the dashboard has a different appearance for each role. This is because the statistics displayed on each role are also different, especially the admin role which uses the dashboard page as a page to monitor news writing management activities. Therefore, the statistics that need to be displayed are much more than other roles.

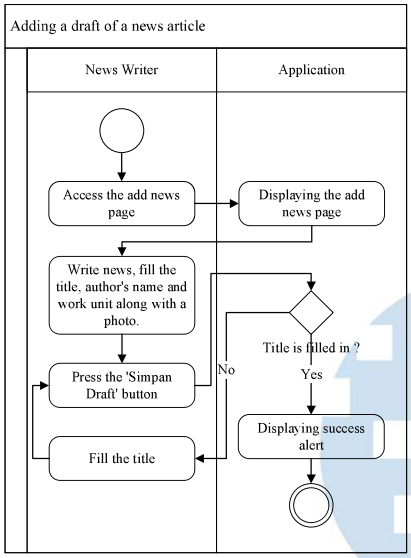

Fig. 10. Activity diagram for adding a draft of a news article of updated process flow

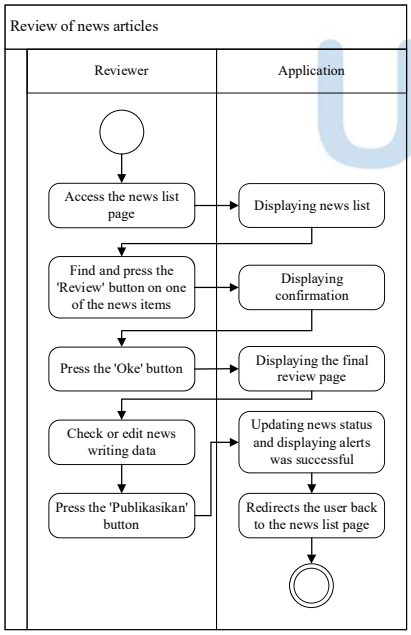

Fig. 11. Activity diagram for reviewing of news article of updated process flow

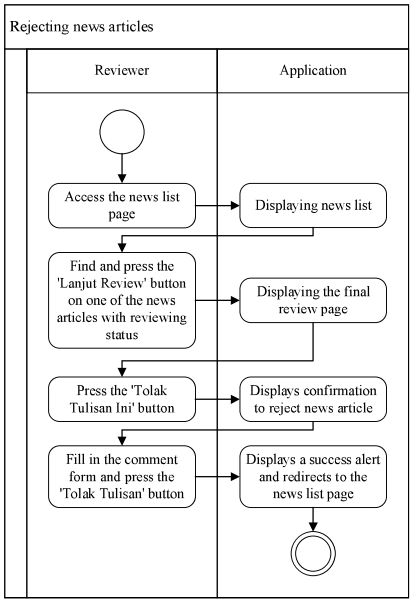

Fig. 12. Activity diagram for rejecting news article (reviewer) of updated process flow

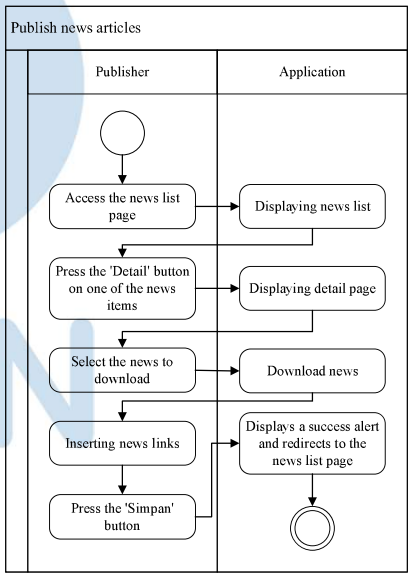

Fig. 13. Activity diagram for publishing news article of updated process flow

In addition to the dashboard page, another page that has changed its appearance is the add news page. According to needs, on the add news page after development, news writers must be able to carry out the news writing process. Therefore, there are changes on that page, where there is a text editor used to write news. Similar to the add news page, the final review page and the news content detail page must be displayed on the page. So it is necessary to redesign the interface. To present the basic scheme of the proposed application display, wireframing techniques are used as a visualization tool.

In this article, only two sample figures are shown from the results of the user interface design that will be applied in the development of this application. Figure 15 is an example of the initial display of the interface design results on the dashboard in the admin role. On the page there is statistical data displayed for the purpose of monitoring news management. Figure 16 is the initial display for the add news page in the news writer role. It has a difference with the previous application, now there is a text editor to write news.

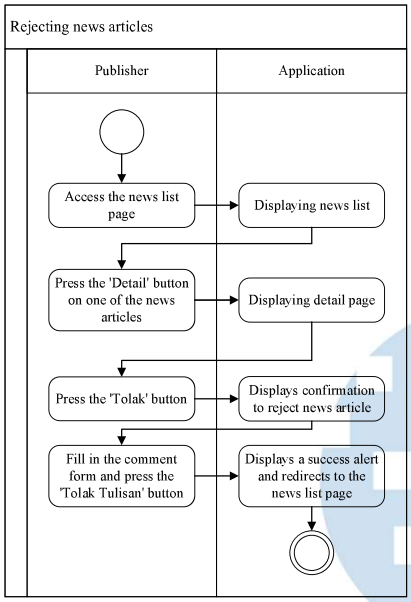

Fig. 14. Activity diagram for rejecting news article (publisher) of updated process flow

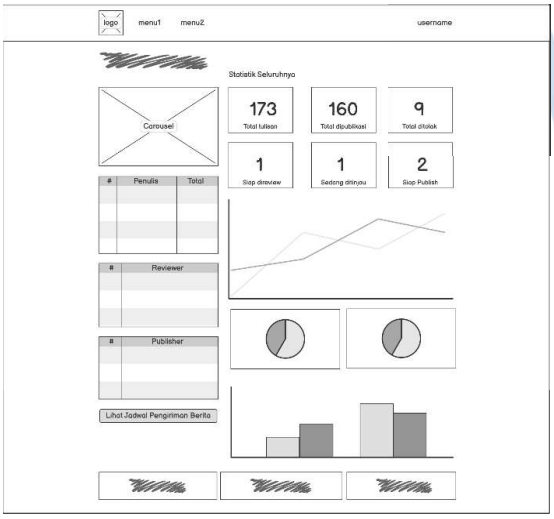

Fig. 15. News monitoring dashboard wireframe

D. Construction

At this stage, the implementation and testing of the news management application that has been carried out in the design stage is explained. Implementation consists of two stages, namely database implementation and program implementation. While testing is carried out using two methods, namely black box and system usability scale.

Fig. 16. Wireframe of page adding news

- Database Implementation

As an initial step in the implementation, modifications were made to the mst_berita table in the previously used news management database. Modifications were made by adding three new fields, namely isi_berita with the TEXT data type, komen with the TEXT data type, and isLihat with the Boolean data type. The addition of these three fields aims to improve the process flow in the application being developed, where previously doc or docx files were used to store news content. With this isi_berita field, news content can be stored in text form in the database so that it is easier to access and manage. The komentar field is needed to store comments from publishers and reviewers in cases where news is rejected so that the author can find out the reason for the rejection. In addition, the isLihat field is used for the notification feature in the application being developed. Meanwhile, the file_draft and file_review fields are not deleted with the aim that users can still access old news articles in the form of files.

- Program Implementation

The implementation of this program includes several parts, namely the development of the application interface display and the implementation of new features proposed at the news management application design stage. After implementing, which uses the codeigniter framework, the role changes and features that can be accessed based on the role can be seen. The application features and roles after development can be seen in Table IX.

- Testing

In the testing phase, the news management application underwent two different types of testing. First, black box testing was carried out to test the functionality of the application. This testing aims to validate whether the application operates according to the specified specifications. Furthermore, testing was carried out using the system usability scale (SUS) to evaluate the level of usability of the application. SUS testing was carried out twice, namely on the previous

application and the latest application, to compare the improvements that had been made.

TABLE IX. FEATURES AND ROLES IN THE UPDATED APPLICATION

Features	Role			
	Writer	Reviewer	Publisher	Admin
Login & logout	✓	✓	✓	✓
Dashboard	✓	✓	✓	✓
Add news	✓			
Edit news	✓	✓		
News draft	✓			
Cancel news	✓			
Notification	✓			
Review news		✓		
Reject news		✓	✓	
Comment on the news		✓	✓	
Fill the news link			✓	
Convert to docx	✓	✓	✓	
Profile	✓	✓	✓	✓
Search news	✓	✓	✓	
News details	✓	✓	✓	
Monitoring				✓
Manage the system				✓
Manage users				✓

For the black box testing process of the news management application that has been developed, the functions to be tested are determined based on functional and non-functional requirements (Tables VI and VII), test scenarios, and the expected results of the test scenarios. All test requirements are written in the black box testing form (Table II). Based on the black box testing that has been carried out, all test scenarios and functions in the news management application have run according to the expected results. All test scenarios have successfully validated the application's functionality, this indicates that the application is able to process data correctly and provide appropriate output.

After the black box testing is completed, it is continued with testing the usability scale of the previous and the latest news management application systems. The results of the comparison of the SUS scores of the news management application before and after development can be seen in Table X.

TABLE X. COMPARISON OF SUS SCORE CALCULATION RESULTS FOR NEWS MANAGEMENT APPLICATIONS

Average SUS Score of News Management Applications	
Old Version	Updated Version
71.3	78.8

Based on testing using SUS, a score of 71.3 was obtained on the previous application and a score of 78.8 on the latest application. From these results, it can be seen that there was an increase in the SUS score of 7.5 points after the development of the news management application. This increase illustrates the improvement and increase in the usability of the application after going through the development process that has been carried out.

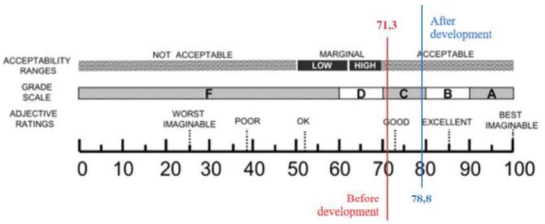

Fig. 17. SUS score comparison results

Based on measurements using a scale developed by Bangor et al. [17] (see Figure 17), in the user acceptance ranges, the previous and latest news management applications received the category "ACCEPTABLE" which means that the both of application version can be well received by users. However, there was a 7.5 point increase in acceptance ranges for the latest application compared to the previous one, indicating a significant improvement in user perception. On the grade scale, both applications have a predicate of "C" or sufficient which shows that the application is not too difficult for respondents to use. But, for the latest applications, it nearly achieved "B" scale or good rating.

There was an increase in the application rating (adjective ratings), where the previous application received a rating of "GOOD" which means that the application has good quality, while the application after development received a rating of "EXCELLENT" which means that the application has very good quality. Although both versions of the application received a predicate of "C" on the grade scale, the change from the rating "GOOD" to "EXCELLENT" in the adjective ratings on the application after development indicates an increase in the quality and performance of the application. This change indicates that development has had a positive impact on improving the user experience and providing a more positive feeling towards the application.

The SUS score for the latest application could potentially be increased, given that respondents only interacted with the application once, during the assessment process. The results could differ if respondents had used the application more than once, allowing us to determine whether the application is easy to understand and use, and whether it offers top-quality results. This allows us to determine what needs to be improved or developed in the future.

E. Deployment

The news management application that has been developed is then implemented on the server of

institution X. In addition, there is a data migration process from files to the database for news data as the application is used. In the news management application, there is a difference in the storage format of news articles between the previous application and the updated application. The previous news management application stores news in Microsoft Word document format (docx), while the updated application stores it in HTML text format. To maintain the existence of data, old document files are not deleted from the application. Therefore, it is necessary to migrate news content from doc or docx files to text so that it can be saved in the database. For the migration scenario, see Figure 18.

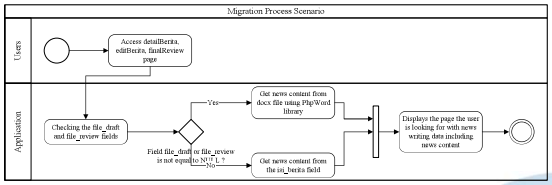

Fig. 18. Migration process scenario

The news content migration scenario occurs when users access news articles via the editBerita, detailBerita, and finalReview pages because on these pages news content in the form of text is needed to be displayed to users. The controller will check the file_review and file_draft fields in the database. If there is content between the two fields, the application will display the news content from the doc or docx files. Conversely, if both fields are empty (null), the application will display the news content stored in the isi_berita field in the database. The process of converting doc and docx files into HTML text using the help of the PhpWord library, then taking paragraphs from the file and saving them into variables containing HTML code which are finally displayed on the editBerita, detailBerita and finalReview pages.

The news management application has been successfully deployed to the X institution server. The data migration process has also been carried out successfully, allowing news data previously stored in files to be accessed and stored in the database. In addition, receiving feedback from application users is an important part of identifying the need for improvements that need to be made to the application. The results of this process provide direction and guidance for further improvement steps to improve the quality and user experience of the news management application.

IV. CONCLUSION

The development of a news management application for institution X using the waterfall model has been successfully carried out. Through a structured development process, the news management application was successfully updated for new features that improve interactivity and performance. All functional requirements of the application have been successfully implemented and run well. Black box testing and system usability scale (SUS) were carried out to test the functionality and usability of the application. The results of functional testing showed that all test scenarios and functions in the updated application ran as expected. In addition, the level of usability of the application showed that the news management application after being developed experienced an average score increase of 7.5 points from the previous score of 71.3 to 78.8. This means that there was an increase in the usability of the application. After development, the news management application got good results from three assessment approaches in usability. On the scale of acceptance, the application was well received by users, showing a significant increase. The assessment scale showed that the news management application after development was considered sufficient and received a very good rating. This shows that application development has a positive impact on the acceptance, feasibility, and quality of the application. Overall, structured work and a clear flow from initiation to end can minimize errors because it has clear details and final descriptions. This is in accordance with the characteristics of the waterfall model in the software development cycle.

REFERENCES

[1] J. H.M., *Analisis dan Desain Sistem Informasi: Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis*. Yogyakarta: Andi Publisher, 2008.

[2] R. A.S and M. Shalahuddin, *REKAYASA Perangkat Lunak: Terstruktur dan Berorientasi Objek*. Bandung: Informatika, 2018.

[3] Roger S. Pressman, *Rekayasa Perangkat Lunak : Pendekatan Praktisi Edisi 7*. Yogyakarta: Andi Publisher, 2012.

[4] R. Rifai and M. Mailasari, "Metode Waterfall pada Sistem Informasi Pengolahan Data Penjualan dan Pembelian Barang," *J. Inform. Univ. Pamulang*, vol. 5, no. 3, pp. 394–401, 2020, doi: 10.32493/informatika.v5i3.6721.

[5] K. Septyanto, M. A. Hamid, and D. Aribowo, "Pengembangan E-Learning Berbasis Website Menggunakan Metode Waterfall," *Elinvo (Electronics, Informatics, Vocat. Educ.*, vol. 5, no. 1, pp. 89–101, 2020, doi: https://doi.org/10.21831/elinvo.v5i1.31054.

[6] N. Purwati, H. B. Pradana, and D. Iswahyuni, "Perancangan Sistem Informasi Berita Berbasis Website PT Garda Revolusi TV Madiun," *CONTEN Comput. Netw. Technol.*, vol. 2, no. 2, pp. 108–117, 2022, doi: 10.31294/conten.v2i2.1670.

[7] Titania Pricillia and Zulfachmi, "Perbandingan Metode Pengembangan Perangkat Lunak (Waterfall, Prototype, RAD)," *J. Ilm. Bangkit Indones.*, vol. 10, no. 1, pp. 6–12, 2021, [Online]. Available: https://doi.org/10.52771/bangkitindonesia.v10i1.153

[8] S. Shaikh and S. Abro, "Comparison of Traditional and Agile Software Development Methodology: a Short Survey," *Int. J. Softw. Eng. Comput. Syst.*, vol. 5, no. 2, pp. 1–14, 2019, doi: https://doi.org/10.15282/ijsecs.5.2.2019.1.0057.

[9] T. Hidayat and M. Muttaqin, "Pengujian Sistem Informasi Pendaftaran dan Pembayaran Wisuda Online menggunakan Black Box Testing dengan Metode Equivalence Partitioning dan Boundary Value Analysis," *J. Tek. Inform. Jutis*, vol. 6, no. 1, pp. 25–29, 2018, [Online]. Available:

https://doi.org/10.33592/jutis.Vol6.Iss1.38

[10] J. Brooke, "SUS : A Retrospective," *J. Usability Stud.*, vol. 8, no. 2, pp. 29–40, 2013, [Online]. Available: https://www.researchgate.net/publication/228593520_SUS_A_quick_and_dirty_usability_scale

[11] K. Khatter and A. Kalia, "Quantification of Non-functional Requirements," in *2014 Seventh International Conference on Contemporary Computing (IC3)*, IEEE, 2014. doi: 10.1109/IC3.2014.6897177.

[12] J. Eckhardt, D. M. Fernandez, and A. Vogelsang, "How to Specify Non-Functional Requirements to Support Seamless Modeling? A Study Design and Preliminary Results," in *2015 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*, Beijing: IEEE, 2015, pp. 164–167. [Online]. Available: 10.1109/ESEM.2015.7321200

[13] R. Ganesh and G. Prabu, "Determination of Internet Banking Usage and Purpose with Explanation of Data Flow Diagram and Use Case Diagram," in *International Journal of Management and Humanities (IJMH)*, Blue Eyes Intelligence Engineering & Sciences Publication, 2020, pp. 52–58. doi: 10.35940/ijmh.G0674.034720.

[14] S. Kendal, *Object Oriented Programming using Java*. London: Bookboon, 2009.

[15] M. T. Ikram, N. A. Butt, A. Hussain, and A. Nadeem, "Testing from UML Design using Activity Diagram: A Comparison of Techniques," *Int. J. Comput. Appl.*, vol. 131, no. 5, pp. 41–47, 2015, doi: 10.5120/ijca2015907354.

[16] U. Salamah and F. N. Khasanah, "Pengujian Sistem Informasi Penjualan Undangan Pernikahan Online Berbasis Web Menggunakan Black Box Testing," *J. Inf. Manag.*, vol. 2, no. 1, pp. 35–44, 2017, [Online]. Available: http://orcid.org/0000-0002-1179-3814

[17] 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, [Online]. Available: https://uxpajournal.org/determining-what-individual-sus-scores-mean-adding-an-adjective-rating-scale/

[18] F. G. Sembodo, G. F. Fitriana, and N. A. Prasetyo, "Evaluasi Usability Website Shopee Menggunakan System Usability Scale (SUS)," *J. Appl. Informatics Comput.*, vol. 5, no. 2, pp. 146–150, 2021, doi: 10.30871/jaic.v5i2.3293.

[19] V. Y. P. Ardhana, "Evaluasi Usability E-Learning Universitas Qamarul Huda Menggunakan System Usability Scale (SUS)," *J. Informatics, Electr. Electron. Eng.*, vol. 2, no. 1, pp. 5–11, 2022, [Online]. Available: https://djournals.com/jieee