International Journal of Scientific Research in Computer Science, Engineering and Information Technology
ISSN : 2456-3307 (www.ijsrcseit.com)
doi : https://doi.org/10.32628/CSEIT22818

Design And Build a School Facility Damage Reporting Application by Applying the Simple Additive Weighting Algorithm

Harni Kusniyati, Muhammad Fadhiil Rachman
Faculty of Computer Science, Mercu Buana University, Jakarta, Indonesia

Article Info
Volume 8, Issue 1
Page Number : 171-179

Publication Issue :
January-February-2022

Article History
Accepted : 15 Jan 2022
Published : 05 Feb 2022

ABSTRACT

There are enough facilities to support teaching and learning activities provided by schools in Indonesia. However, facilities damage can disturb the learning activities at school. To maintain the eligibility of school facilities, it is necessary to report facilities damage by submitting complaint to the school administration staff. However, there is a new problem during the process of the damage reporting, namely slow response to the report that has been submitted by students. Therefore, it is important to have a reporting application that is accessible for both students and school administration staff. It is expected that the school be able to respond to students' complaints quickly in order to ensure the continuity of learning activities at school. Simple Additive Weighting (SAW) algorithm is able to provide accurate decision by prioritizing based on the criteria for the severity of school facilities damage. The report can be responded immediately by supporting the best report of all school facilities damage reports that is submitted by students through web application, which is selected by SAW algorithm decision.

Keywords: *Simple Additive Weighting, reporting application, school facilities.*

I. INTRODUCTION

Nowadays, school services in Indonesia have been something common in the society, in which such condition may increase the level of satisfaction of the users of education, in order to support the continuity of learning activities at school. The quality of school is related to a good service performance offered by the school. High quality of school will determine the right selling price. Therefore, a good service performance will increase the selling price.

In fact, some schools still rely on manual method instead of moving on new technology. The quality of school, in order to speed up and to smooth the access to school services, can be increased by creating an innovation to help overcome every problem at school. Problems commonly occurring at school are school administration and the eligibility of school facilities during teaching and learning activities.

One of the schools in Jakarta is Public High School 29 of Jakarta. The facilities owned by the school are classrooms and computer laboratory. Facility becomes one of the students' needs to support their comfort during teaching and learning activities. Therefore, facility is an important thing that the school should always concern. To ensure the comfort during teaching and learning activities, facilities damage becomes one of the problems that should be overcome [1].

This school has a service procedure from administration staff, such as reporting facilities damage, but it is still done manually in the form of a written report. As a result, some problems occur, namely slow response to the facilities damage report, errors in data record of damage report, and the lack of control as well as supervision on the work administration staff.

To overcome the problems above, this research will build a service application that is accessible for the students and administration staff. Thus, the process of responding to the report of school facilities damage can run quickly and properly.

A research on the same field is conducted by Rusmi [2]. The research that was conducted in Andalas University of Padang obtained SMS-Gateway-based information system for damage report of facilities and infrastructure. In this research, the method used is Waterfall development method using Unified Modeling Language (UML).

Other similar research is conducted by Ferdiansyah [1]. This research obtains a reporting application, which is made based on Android and Web, for facilities and infrastructure damage at STT Nurul Hadid. In this research, Waterfall method is used as the development of its application system.

Another similar study is carried out by Permatasari [3]. This research gains web-based reporting application for the community. This research uses Waterfall method in developing software and UML diagram as the system designing.

Reporting application is software used for submitting complaints of a problem in order to get response, in the form of a follow-up to the complaint, from the staff in charge. Considering some problems occurring, the researcher carries out a number of studies on the mechanism of reporting school facilities damage by using an algorithm to speed up the process of selecting a large number of reports, which is Simple Additive Weighting.

Simple Additive Weighting (SAW) is also called weighted sum method. The basic concept of SAW method is to find the weighted sum of the performance rating for each alternative on all attributes [4]. The results of SAW calculation obtain value of each report selectively done by the school administration staff in selecting the highest-rank report. Thus, it is expected that the use of this technology help the school administration staff to handle the school facilities damage report.

II. METHODS AND MATERIAL

A. *Simple Additive Weighting*

The basic concept of this algorithm is to find the average value obtained from the calculation of each alternative by multiplying the scale value given to the attribute of an alternative with the important weight of the attribute given by the decision maker which then sums the object's value for all criteria [5]. The method that applies the weighted sum with the concept of finding alternatives by calculating the weight of each criterion on all attributes is called Simple Additive Weighting (SAW) method [6]. SAW method is recommended to fix the selection problems in multiprocessing decision making system [7], where each expert evaluates alternatives to performance criteria [8].

The advantage of using *Simple Additive Weighting* method is its accuracy on scoring the evaluation because it is based on a predetermined weight value of each attribute, that will then proceed with the ranking process based on the best alternative obtained from a number of other alternatives [5].

According to Irvanizam [9], *Simple Additive Weighting* calculates the evaluation score of each alternative by multiplying the scale value. This method also requires the process of normalization of the decision matrix (X) to a scale that is proportional to all existing alternative ranking [10].

The basic concept of *Simple Additive Weighting* [11]:

$$r_{ij} = \begin{cases} \dfrac{x_{ij}}{max_i\, x_{ij}} & If\ j\ is\ the\ benefit\ atribute \\ \dfrac{min_i\, x_{ij}}{x_{ij}} & if\ j\ is\ the\ cost\ atribute \end{cases}$$

Information:

r_{ij} = Normalized report rating value

x_{ij} = Attribute value owned by each criterion

max_i = The biggest value of each criterion i

min_i = The smallest value of each criterion i

$benefit$ = If the biggest value is the best

$cost$ = If the smallest value is the best.

Where rij is normalized rating report from alternative Ai on atribute Cij i=1,2,...,n. The preference value for each alternative (Vi) is given as:

$$V_i = \sum_{j=1}^{n} w_j r_{ij}$$

Information:

V_i = Ranking for each alternative

w_j = Rank weight value (of each criterion)

r_{ij} = Normalized report rating value

The higher value Vi identifies that the alternative Ai is preferred.

B. Waterfall

This research applies the software development model of Waterfall. According to Rizhul [12], waterfall model is a systematic and sequential approach starting from the level of system requirement analysis, proceeding to stage of planning, modeling, construction, and implementation of the software system to the use. It always ends with application support. Various stages of this model are:

[1] Requirement Analysis

This stage is an analysis for the needs of application and steps to collect data through data observation or interview, and the data is collected using case-based reasoning method. System service, constraints, and goals are determined by the results of consultation with users, which are then defined in detail and used as system specification.

[2] Application Design

The stage of system design allocates the system requirements, both hardware and software, by forming an overall system architecture. Software design involves the stage of identifying and describing the software system's basic abstraction and its correlation.

[3] Implementation

In this stage, software design is realized as a series of programs or program units. Testing involves the process of verifying that each unit meets its specifications.

[4] Testing

The individual units of the program or programs are combined and tested as a complete system to ensure whether they match the software requirements. After testing, the software can be sent to the customer.

[5] Maintenance

Usually (although not always), this stage is the longest one. The system is installed and used significantly. Maintenance involves the process of correcting errors that are not found in the previous stages, increasing the implementation of the system unit, and improving system services as new requirements.

B. Data *Collection* Method

In this research, data and information are needed to analyze the problems and design. The methods used are:

1) Requirement Analysis

This observation method is carried out directly to observe the various activities done by students and employees in the administration division in Public High School 29 of Jakarta and to find out more about the application requirements that will be needed.

2) Interview

The researcher conducted a direct interview with one of the employees of the school administration division who handled the reporting of school facilities damage. The interview was about the process of reporting service process and facilities damage report in Public High School 29 of Jakarta, along with the staff in other division that could provide relevant information in this study.

3) Case-based Reasoning

According to Sri [13], *Case-based Reasoning* is a method to overcome problems by using the previous

experience based on the similarity to one or some other solutions.

This method uses case-based reasoning knowledge to know the process of problem solving in facilities damage report by mapping the basic supporting criteria of problem solving from the solutions of the past similar problems in school report.

C. Unified Modeling *Language*

This method uses UML (*Unified Modeling Language*) in designing a reporting application for school facilities damage. UML is one of the standards of language commonly used in industrial field to define the need, to make an analysis and design, and to depict architecture in object-oriented programming [14].

UML is included in graphical notation supported by single model, which help describe and design software system, especially a system built using object-oriented programing (OOP) [15].

The actor-definition table below explains details of actors involved in the reporting application for school facility damage.

TABLE I. DEFINITION OF ACTORS

Actor	Weight
Administration	This actor can manage data of students, rooms, facilities, and damage type. This actor also processes and summarizes the reports.
Students	This actor can act as a reporter of damage by submitting a damage report. This actor can also view the report history that has been made.

4) Activity Diagram

Activity diagram is modeling the workflow of a business process and the sequence of activities in a process [15]:

III. RESULTS AND DISCUSSION

A. Analysis on Criteria

This is the stage in the process of selecting the best report of all reports made by the students. The selected report has criteria formula that has been analyzed as a reference on the calculation of Simple Additive Weighting algorithm.

Weight criteria in SAW method are determined by the decision makers. In other words, policy makers must decide weight preferences in the beginning for each criterion [16].

Some of the required criteria are mentioned in these tables:

TABLE I. CRITERIA ATTRIBUTE

Criteria	Benefit	Cost
Very low	✓	-
Low	✓	-
Medium	✓	-
High	✓	-
Very high	✓	-

TABLE II. CRITERIA WEIGHT

Code	Criteria	Weight
C1	Very low	0.1 (10%)
C2	Low	015 (15%)
C3	Medium	0.2 (20%)
C4	D. High	0.25 (25%)
C5	Very high	0.3 (30%)

TABLE III. CRITERIA VARIABLE

Variable	Value
Never	1
Only once	10
Seldom	40
Often	50

B. Sample Data of Report

Sample data is taken from reports that have been made by students as a reference to get the best report so that the administration can get the best report calculated using the SAW algorithm.

TABLE IV

SAMPLE DATA OF REPORT

Core of Report (Alter-native)	Criteria				
	C1	C2	C3	C4	C5
L1	Seldom	Never	Never	Never	Never
L2	Never	Never	Only Once	Never	Seldom
L3	Never	Only Once	Never	Often	Never
L4	Only Once	Never	Seldom	Never	Only Once
L5	Never	Often	Seldom	Seldom	Never
L6	Never	Only Once	Only Once	Never	Only Once
L7	Seldom	Often	Never	Only Once	Never
L8	Often	Often	Only Once	Never	Never
L9	Never	Often	Only Once	Only Once	Never
L10	Only Once	Seldom	Never	Never	Often

C. Match Rating

From the sample data in the previous section, the criteria variable in the report are then matched with the results of the analysis on the criteria variable in table 4, and then changed into the value of the variable. Look at the table below:

TABLE V. REPORT MATCH RATING

Core of Report (Alternative)	Criteria				
	C1	C2	C3	C4	C5
L1	40	1	1	1	1
L2	1	1	10	1	40
L3	1	10	1	50	1
L4	10	1	40	1	10
L5	1	50	40	40	1
L6	1	10	10	1	10
L7	40	50	1	10	1
L8	50	50	10	1	1
L9	1	50	40	10	1
L10	10	40	1	1	50

D. Report Matrix (X)

Report matrix (X) is made based on table 6, which is the match rating of each criterion. Values of the results of the table are then made into matrix as seen below:

$$X = \begin{pmatrix} 40 & 1 & 1 & 1 & 1 \\ 1 & 1 & 10 & 1 & 40 \\ 1 & 10 & 1 & 50 & 1 \\ 10 & 1 & 40 & 1 & 10 \\ 1 & 50 & 40 & 40 & 1 \\ 1 & 10 & 10 & 1 & 10 \\ 40 & 50 & 1 & 10 & 1 \\ 50 & 50 & 10 & 0.5 & 1 \\ 1 & 50 & 40 & 10 & 1 \\ 10 & 40 & 1 & 1 & 50 \end{pmatrix}$$

E. Normalization Matrix (R)

According to Ciptayani [17], normalization is done to get the weight of each criterion, and then SAW method will be carried out to obtain a decision. Report normalization matrix performs formula calculation for criteria based on the criteria attribute found in the criteria analysis in table 3 and the type of report criteria attribute. Thus, the normalization matrix R is obtained as follows:

$$R = \begin{pmatrix} 0.8 & 0.02 & 0.025 & 0.02 & 0.02 \\ 0.02 & 0.02 & 0.25 & 0.02 & 0.8 \\ 0.02 & 0.2 & 0.025 & 1 & 0.02 \\ 0.2 & 0.02 & 1 & 0.02 & 0.2 \\ 0.02 & 1 & 1 & 0.8 & 0.02 \\ 0.02 & 0.2 & 0.25 & 0.02 & 0.2 \\ 0.8 & 1 & 0.25 & 0.2 & 0.02 \\ 1 & 1 & 0.25 & 0.02 & 0.02 \\ 0.02 & 1 & 1 & 0.2 & 0.02 \\ 0.2 & 0.8 & 0.025 & 0.02 & 1 \end{pmatrix}$$

F. Ranking

Ranking process aims at looking for the values of normalized matrix that is processed by ranking formula. The following formula is used to determine value V1 until V10:

$$V1 = (0.10 \times 0.8) + (0.15 \times 0.02) + (0.20 \times 0.025) + (0.25 \times 0.02) + (0.30 \times 0.02)$$
$$= 0.08 + 0.003 + 0.005 + 0.005 + 0.006$$
$$= 0.099 = 0.099$$

$$0.10 \times 0.02 + 0.15 \times 0.02 + 0.20 \times 0.25 + 0.25 \times 0.02 + 0.30 \times 0.8$$
$$= 0.002 + 0.003 + 0.05 + 0.005 + 0.24 + 0.24$$
$$= 0.3$$

$$V3 = (0.10 \times 0.02) + (0.15 \times 0.2) + (0.20 \times 0.025) + (0.25 \times 1) + (0.30 \times 0.02)$$
$$= 0.002 + 0.03 + 0.005 + 0.25 + 0.0066$$
$$= 0.293$$

$$V4 = (0.10 \times 0.2) + (0.15 \times 0.02) + (0.20 \times 1) + (0.25 \times 0.02) + (0.30 \times 0.2)$$
$$= 0.02 + 0.003 + 0.2 + 0.005 + 0.06$$
$$= 0.288 = 0288$$

$$V5 = (0.10 \times 0.02) + (0.15 \times 1) + (0.20 \times 1) + (0.25 \times 0.8) + (0.30 \times 0.02)$$
$$= 0.002 + 0.15 + 0.2 + 0.2 + 0.006$$
$$= 0.002 + 0.15 + 0.2 + 0.2 + 0.006$$

$$V6 = (0.10 \times 0.02) + (0.15 \times 0.2) + (0.20 \times 0.25) + (0.25 \times 0.02) + (0.30 \times 0.2)$$
$$= 0.002 + 0.03 + 0.05 + 0.005 + 0.06.06$$
$$= 0.147$$

$$V7 = (0.10 \times 0.8) + (0.15 \times 1) + (0.20 \times 0.25) + (0.25 \times 0.02) + (0.30 \times 0.02)$$
$$= 0.08 + 0.15 + 0.005 + 0.05 + 0.0066$$
$$= 0.291$$

$$V8 = (0.10 \times 1) + (0.15 \times 1) + (0.20 \times 0.25) + (0.25 \times 0.02) + (0.30 \times 0.02)$$
$$= 0.1 + 0.15 + 0.05 + 0.005 + 0.006$$
$$= 0.311 = 0311 = 0.311$$

$$0.10 \times 0.02 + 0.15 \times 1 + 0.20 \times 1 + 0.25 \times 0.2 + 0.30 \times 0.02$$
$$= 0.002 + 0.15 + 0.2 + 0.05 + 0.006$$
$$= 0.002 + 0.15 + 0.2 + 0.05 + 0.006$$

$$V10 = (0.10 \times 0.2) + (0.15 \times 0.8) + (0.20 \times 0.025) + (0.25 \times 0.02) + (0.30 \times 1)0 = 0.10 \times 0.2 + 0.15 \times 0.8 + 0.20 \times 0.025 + 0.25 \times 0.02 + 0.30 \times 1$$
$$= 0.02 + 0.12 + 0.005 + 0.005 + 0.30.3$$
$$= 0.45$$

After being calculated using the formula, the value is sorted. The list of alternative decisions order resulting from the calculation using SAW method starts from the largest value [18].

TABLE VI

RESULTS OF RANK VALUE

No	Code of report	Values
1	L5	0.558
2	L10	0.45
3	L9	0.408
4	L8	0.311
5	L2	0.3
6	L3	0.293
7	L7	0.291
8	L4	0.288
9	L6	0.147
10	L1	0.099

According to the results seen in Table 7 calculated using Simple Additive Weighting (SAW), report that has a high damage is the one with code of report of L5, with the damage value of 0.558.

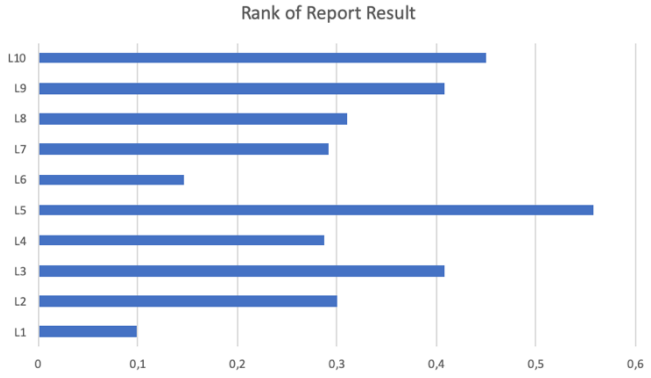

Figure 1.Graph of rank value

G. Implementation of Application

This page can only be accessed by students, which is used to make a school facilities damage report.

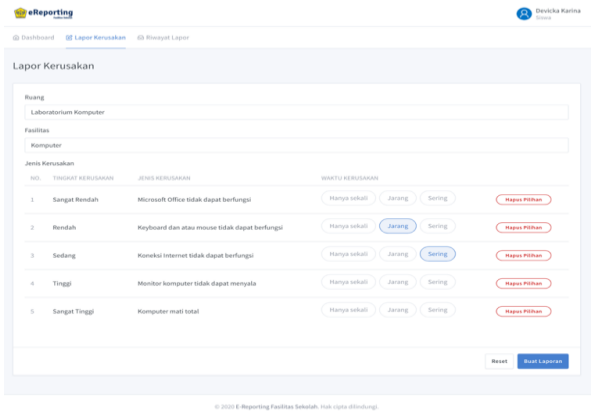

Figure 2. Student Page of Damage Report

Function of report process page is to help administration staff to process the facilities damage report that has been made by students as well as report that has not been processed before using this reporting application for facilities damage. In this page, process of *Simple Additive Weighting* algorithm is used to get a report with high severity of damage of all reports that have been selected by administration staff.

The next stage after implementing the application is testing Black Box. According to Manish [19], Black Box testing is a test done without knowing the internal working of the application being tested. Black Box Testing Method is a method that is easy to use because it only requires the lower and upper limits of the expected data [20]. Thus, this test produces the desired output without looking at internal processes or program code executed by the software [21]. By testing the Black Box, it can be seen whether the application that has been made is running well [22] [23]. The results of the analysis on Black Box testing in this application are seen below:

TABLE VII

TESTING RESULT OF BLACK BOX

Testing object	Explanation	Testing results
Report process testing	Testing results of the report processing done by administration staff can be said to be successful. The application shows a report notification that has been successfully processed, and the processing facilities damage report data has been updated.	Matched
Report record testing	The report record page that is accessed by the administration staff and options on this page that are modified show the report record data. It means that this test runs well.	Matched
Damage report testing	Testing results done by students show a notification that the report is received successfully. Thus, this test succeeds in making a new facilities damage report, which is then stored in a database.	Matched
Report history page testing	The report history page accessed by students shows the report history data in the table. It means that this test runs well.	Matched

IV. CONCLUSION

Based on the results of the implementation of the application built, there are some conclusions, namely:

1) This application is able to help school administration in reducing the processing time of recording school facilities damage reports.
2) This application is able to replace the manual system that has been running so far in reporting school facilities damage, especially damage reports about classrooms and computer laboratories.

V. REFERENCES

[1]. M. S. Ferdiansyah, M. Jasri, and Widijianto, "Aplikasi Quick Response Dalam Melayani Pengaduan Kerusakan Sarana STT Nurul Jadid Berbasis Android Dan Web," Pros. SENTIA, vol. 8, pp. 152–157, 2016.

[2]. R. Rusmi, T. A. Sundara, and Afrinaldi, "Sistem Informasi Pelaporan Kerusakan Sarana Dan Prasarana pada Asrama Mahasiswa Universitas Andalas Padang," STMIK Indones. Padang, vol. 8, no. 1, pp. 78–90, 2019.

[3]. R. A. Permatasari, B. Priyambadha, and A. Arwan, "Pengembangan Sistem Aplikasi Pelaporan Masyarakat Berbasis Web di Kabupaten Pekalongan," J-PTIIK, vol. 2, no. 11, pp. 5604–5610, 2018.

[4]. Y. E. Chintyari and T. Prihatin, "Implementasi Metode Simple Additive Weighting Untuk Pemilihan Guru Berprestasi Pada SMP Islam Pondok Duta," JITK, vol. 3, no. 2, pp. 133–138, Feb. 2018.

[5]. A. P. U. S. Khairul, Manogari Simaremare, "Decision Support System in Selecting The Appropriate Laptop Using Simple Additive Weighting," Int. J. Recent Trends Eng. Res., vol. 2, no. 12, pp. 215–222, 2016.

[6]. L. P. Wanti et al., "A support system for accepting student assistance using analytical hierarchy process and simple additive weighting," in Journal of Physics: Conference Series, 2020, vol. 1430, p. 12034.

[7]. R. Sahara, Y. Jumaryadi, and A. Kartika, "Decision Support System For The Best Teacher Election With Simple Additive Weighting Method Based On Web (Case Study On Al-Ijtihat Vocational School)," Int. Res. J. Comput. Sci., vol. 5, no. 03, pp. 103–110, 2018.

[8]. D. Borissova, D. Korsemov, and I. Mustakerov, "Multi-Criteria Decision Making Problem for Doing Business: Comparison Between Approaches of Individual and Group Decision Making," in Computer Information Systems and Industrial Management, 2019, vol. 11703, pp. 385–396.

[9]. I. Irvanizam, "Multiple attribute decision making with simple additive weighting approach for selecting the scholarship recipients at Syiah Kuala university," in 2017 International Conference on Electrical Engineering and Informatics (ICELTICs), 2017, pp. 245–250.

[10]. E. Siswanto and A. W. Katili, "Implementation of decision support system for campus promotion management using fuzzy multiple analytic decision making (FMADM) method (Case study: Universitas multimedia nusantara)," in 2017 4th International Conference on New Media Studies (CONMEDIA), 2017, pp. 75–80.

[11]. R. Wati and E. Mayasari, "Sistem Pendukung Keputusan Pemilihan Bibit Sapi Unggul dengan Metode Simple Additive Weighting (SAW) pada Peternakan Sapi Sriagung Padangratu Lampung Tengah," J. TAM (Technology Accept. Model., vol. 5, no. 4, pp. 22–28, 2015.

[12]. Y. S. T. Rizhul Januar Ramadhan, "Application Design Of The National Book Reader Using Zahman Framework Method," Int. Res. J. Comput. Sci., vol. 5, no. 04, pp. 168–178, 2018.

[13]. E. W. Sri Mulyana, Sri Hartati, Retantyo Wardoyo, "Case-Based Reasoning for Selecting Study Program in Senior High School," Int. J. Adv. Comput. Sci. Appl., vol. 6, no. 4, pp. 136–140, 2015.

[14]. R. Sahara, H. Prastiawan, and D. Rizal, "Rancang Bangun Sistem Informasi Mylibrary Telkomsel Berbasis Website (Studi Kasus: PT. Telekomunikasi Selular)," FORMAT, vol. 6, no. 1, pp. 106–118, 2017.

[15]. G. W. Sasmito, "Penerapan Metode Waterfall Pada Desain Sistem Informasi Geografis Industri Kabupaten Tegal," JPIT, vol. 2, no. 1, pp. 6–12, 2017.

[16]. A. Ibrahim and R. A. Surya, "The Implementation of Simple Additive Weighting (SAW) Method in Decision Support System for the Best School Selection in Jambi," in Journal of Physics: Conference Series, 2019, vol. 1338, p. 12054.

[17]. P. I. Ciptayani and K. C. Dewi, "Decision support system of e-book provider selection for library using Simple Additive Weighting," J. Phys. Conf. Ser., vol. 953, p. 12066, Jan. 2018.

[18]. C. B. Slamet Kurniawan Fahrurozi, Dwi Maryono, "Simple Additive Weighting Method in the Development of a System Assessing the Feasibility of Job Training Industry," Indones. J. Informatics Educ., vol. 1, no. 2, pp. 17–28, 2017.

[19]. Manish Kumar, S. K. Singh, and D. R. K. Dwivedi, "A Comparative Study of Black Box Testing and White Box Testing Techniques," Int. J. Adv. Res. Comput. Sci. Manag. Stud., vol. 3, no. 10, pp. 32–44, 2015.

[20]. M. S. Mustaqbal, R. F. Firdaus, and H. Rahmadi, "Pengujian Aplikasi Menggunakan Black Box Testing Boundary Value Analysis (Studi Kasus : Aplikasi Prediksi Kelulusan SNMPTN)," JITTER, vol. I, no. 3, pp. 31–36, 2015.

[21]. D. Febiharsa, I. M. Sudana, and N. Hudallah, "Uji Fungsionalitas (BlackBox Testing) Sistem Informasi Lembaga Sertifikasi Profesi (SILSP) Batik Dengan AppPerfect Web Test Dan Uji Pengguna," JOINED J., vol. 1, no. 2, pp. 117–126, 2018.

[22]. Y. J. Suryani Dewi, Linda Miftahul Jannah, "Analisis Dan Perancangan Sistem Informasi Manajemen Aset Tetap Pada PT. Metis Teknologi Corporindo," J. Sist. Informasi, Teknol. Inf. dan Komput., vol. 9, no. 1, pp. 81–91, 2018.

[23]. N. Setiyawati and S. Teguh Bowo Kesuwo, "Pembangunan Aplikasi Pelaporan Kecelakaan Lalu Lintas Berbasis Web Menggunakan Framework Laravel," Assoc. Inf. Syst. – Indones., vol. 2, no. 22, pp. 1–10, 2017.

Cite this article as :

Harni Kusniyati, Muhammad Fadhiil Rachman, "Design And Build a School Facility Damage Reporting Application by Applying The Simple Additive Weighting Algorithm", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 8 Issue 1, pp. 171-179, January-February 2022. Available at doi : https://doi.org/10.32628/CSEIT22818
Journal URL : https://ijsrcseit.com/CSEIT22818