Journal of Computer Science

Original Research Paper

Decision Support Model for Determining the Best Employee using Fuzzy Logic and Simple Additive Weighting

Lucky Christopher Chen and Ditdit Nugeraha Utama

Computer Science Department, Bina Nusantara University, Jakarta, Indonesia

Article history
Received: 15-11-2021
Revised: 22-03-2022
Accepted: 16-06-2022

Corresponding author
Lucky Christopher Chen
Computer Science Department,
BINUS Graduate Program -
Master of Computer Science,
Bina Nusantara University,
Jakarta, Indonesia 11480
Email: lucky.chen@binus.ac.id

Abstract: Specifically, for an information technology company, identifying the best employees (the best programmers) is valuable. The election is intended to improve the performance of the company's programmers. The company's performance will improve as programmers' performance improves. This study attempts to develop a Decision Support Model (DSM) to identify the best employees (i.e., programmers) in the firm. The model considered nine parameters (technical skills, problem-solving, communication skills, teamwork, discipline, work progress, time management, formal education, and informal education) by integrating the Fuzzy Logic (FL) with the Simple Additive Weighting method (SAW). This model is finally able to be benefitted by information and technology firms. This can eventually be used to assess and view employee evaluations, making it simpler for businesses to make key decisions (e.g., granting incentives, salary raises, or promotions). The model is based on data collected from ten company programmers (where six of them are real data). The model concludes that programmer 2 is the best employee in the firm, with a total score of 97.94, based on the suggestions from the constructed DSM.

Keywords: Decision Support Model, Fuzzy Logic, Simple Additive Weighting, Best Employee, Programmer

Introduction

The development of a business may be influenced by several factors, one of which is the company's human resources or employees. Employees are residents of working age "aged 15-64 years" or the total population in a nation that produces products and services if there is a demand for their job and if they want to participate in the activity, according to Manulang (2002). In that general context, an employee is someone who provides "mind or energy" services for a fixed amount.

Employees have a significant effect on a company's ability to expand since they are critical to the success of all activities inside the organization Bashir and Ramay (2010). Each employee has a distinct set of talents and abilities. The best personnel can be selected in a variety of ways, including direct selection by the manager or firm leadership. However, this strategy is viewed as unfair since it lacks a clear outcome in the form of computing points, or scores based on employee performance. Employees have several concerns about this strategy, including the degree of accuracy, standards, or evaluation criteria and the time problem in picking the best employee in the firm (Yaqin *et al*., 2014; Faisol *et al*., 2014)

In this study, we created a Decision Support Model (DSM) to help a company determine the best employee that focuses on programmers. To decide on the best programmers in a firm, certainly, reasonable calculations must be performed so that the conclusions may be accepted by all programmers. The goal of finding the best programmers is to offer a metric for gauging promotions, pay increases, or other incentives that programmers will get. The development of this DSM will benefit the company by providing more accurate results, saving time, improving company performance, improving programmers' performance, and making it easier for companies to determine the best programmers (Anindita, 2020; Yusuf *et al*., 2020; Vosloban, 2012; Yuliani, 2013).

Several prior studies proposed DSM using the usual Fuzzy Logic (FL) method. Several studies that employed this method did not attain maximum results, or the method's results were deemed less accurate. As a result, several studies have been conducted to develop DSM by merging the FL method with other methods (e.g., Profile Matching/PM, Analytical Hierarchy Process/AHP, and others).

Many DSM-related types of research employ the main notion of FL to evaluate human resource performance. Yaqin *et al*. (2014) used the fundamental notion of FL to choose a thesis supervisor who met the student's criteria.

Yuliani (2013) employed the AHP to find the best staff at KFC. Nardiono (2017) researched to select the best workers by comparing two methods, namely Simple Additive Weighting (SAW) and Weighted Product (WP).

The decision-making process must be focused on the importance of the finest criteria that each programmer must possess. All of this necessitates the use of a DSM to aid in the selection of the ideal programmers. In this study, we looked at nine validated characteristics and used a mix of FL idea and the SAW method. Technical abilities, communication skills, problem-solving, time management, discipline, teamwork, job development, and formal and informal education are among the nine factors. The DSM was built on a website to make it easier for businesses to submit programmers' points for each aspect.

Literature Review

Decision Support Model (DSM)

DSM is a model that can provide problem-solving or decision support used by organizations, individuals, companies, and others to get maximum decisions. The concept of a decision support system was first expressed by Scott Morton (1970) who explained that DSM is a computer-based system intended to assist decision-makers in utilizing certain data and models to solve various unstructured problems Daihani (2001).

There are several important components (Fig. 1) that must be owned in a DSM Turban *et al*. (2011) including:

1. Data Management Subsystem (database): DSM component that is useful as a data provider for the system. The data is stored and organized in a database organized by a system called a database management system
2. Model management subsystem (model base): The uniqueness of DSM is its ability to integrate data with decision models. The model is an imitation of the real world
3. Dialogue management subsystem (user interface): The facility that can integrate the installed system with the user interactively, is known as the dialogue subsystem. Through the dialogue subsystem, the system is implemented so that the user can communicate with the created system

Many DSM-related types of research employ the main notion of FL to evaluate human resource performance. Yaqin *et al*. (2014) employed the fundamental notion of FL to choose a thesis supervisor who met the student's requirements. The lecturer's area of competence, the student's GPA, the load of the lecturer's advice, the duration of the guidance, and the worth of the thesis are all used in this study. The testing data is drawn from thirty historical records chosen at random from a total of 2,163 records. According to the study's findings, an accuracy rate of 87% was reached when compared to the prior manual appointment. You may combine the FL method with additional ways to increase the degree of precision.

Susilo (2017) conducted subsequent research on the DSM method with the application of PM for the selection of the study's head. This method compares individual capabilities to job competencies so that disparities in competencies may be identified as gaps. The smaller the resultant gap, the higher the value weight, which indicates it has a better chance of becoming the study program's leader. There are also several areas of evaluation, such as personality, social, and planning. To determine the worth of this component, a questionnaire will be sent and filled out by other lecturers. Using this procedure, the highest overall score is utilized as a recommendation from the study program's director.

Utama and Rustamaji (2018) constructed a DSM using a variety of methods, including object-oriented, hill climbing, survey, and FL. Each of these strategies is used to assess and develop models, justify and analyze data, maximize proposed decisions, and gather data. The four key factors considered in this study are education, teaching ability, research, and social responsibility. As data sources, 35 teachers from one institution in Jakarta were chosen. According to the study's findings, using this method can save 47.5% of the time while producing DSM.

Furthermore, Fitriana *et al*. (2018) addressed the DSM methods in conjunction with the use of PM and the system development life cycle in the process of choosing excellent students at the Faculty of Engineering. Dummy data is utilized as a data source in PM computation scenarios. This research also employs black box testing to ensure that the input/output data is consistent with expectations. In addition, a User Acceptance Test (UAT) was performed to demonstrate that the DSS had aided in decision-making and that certain adjustments were needed to make the system even better.

Utama and Oktafiani (2020) conducted a study in which they compared the DSM method to the usage of Profile Matching to identify the best marketers in a Jakarta-based health firm. This study included seventeen criteria, which were divided into three core factors and fourteen secondary factors. The whole final score was then computed and graded from the highest (the best marketer) to the lowest value (the worst marketer).

Yusuf *et al*. (2020) investigated the best front-end framework for desktop-based web applications. This study's goal is to provide a framework that is simpler and easier to utilize. SAW was utilized for this research, with five requirements in mind: Preprocessor, responsive, browser support, easy to use, and template. Based on the evaluation outcomes, research may be developed by incorporating more criteria and compared to other methodologies.

Fuzzy Logic (FL)

FL is a logic with an ambiguity or fuzziness value between true and false. A bias value is either true or false in FL theory. However, the size of an object's existence and inaccuracy is determined by the weight of its membership. FL has membership degrees ranging from 0 to 1. In contrast to traditional logic, which only offers two possible outcomes: 1 or 0 Zadeh (1965). FL is used to convert a quantity stated by language (linguistics), such as the amount of vehicle speed, which is expressed slowly, rather fast, fast, and very fast. And FL shows the extent to which a value is true and the extent to which a value is false.

FL is frequently used in circumstances including uncertainty, imprecision, and other variables. FL is a language that blends precise machine language with meaning-focused human language. FL was made utilizing human natural language. To express an idea, Zadeh (1975) used the phrase "Notion of Linguistic Variables".

In FL, there is an algorithm (Fig. 2) that is used to obtain a precise value or what is known as crisp output Utama (2021). A fuzzification operation will be performed on the raw data stored (the process for converting data into fuzzy values). Then begin the de-fuzzification process, which seeks to convert the fuzzy value into a definite one (crisp output).

Simple Additive Weighting (SAW)

SAW is sometimes referred to as a weighted sum method of decision-making. The procedure of normalizing the choice matrix to a scale that can be utilized with all ratings of the available options is required for this method. SAW is a popular strategy for solving Multiple Attribute Decision Making (MADM) issues. MADM is a method for selecting the best option from a set of alternatives based on certain criteria Sahir *et al*. (2017).

In solving problems using the SAW method, several steps need to be done, namely:

1. Determine the criteria that will be used as a reference in making decisions
2. Determine the suitability rating or weight for each criterion that we make
3. Create a decision matrix derived from the criteria that have been made
4. Normalization of the matrix described in Eq. (1) is based on the equation that is adjusted to the type of attribute to get a normalized matrix. To get the normalization value (r_{ij}) is calculate the criteria value (x_{ij}) divided by the maximum value of each criterion (benefit criteria) and the minimum value of the criteria divided by the value of each attribute (cost criteria)

$$r_{ij} = \begin{cases} \dfrac{x_{ij}}{MAX\left(X_{ij}\right)}(benefit) \\ \dfrac{MIN_i\left(X_{ij}\right)}{X_{ij}}(\cos t) \end{cases} \tag{1}$$

5. The result is obtained by calculating the multiplication of the normalized value (r_{ij}) with the weight value (W_j) or the formula can be seen in Eq. (2). Then the value of each criterion will be added up. The alternative with the highest score is the winner.

$$V_i = \sum_{j=1}^{n} W_j r_{ij} \tag{2}$$

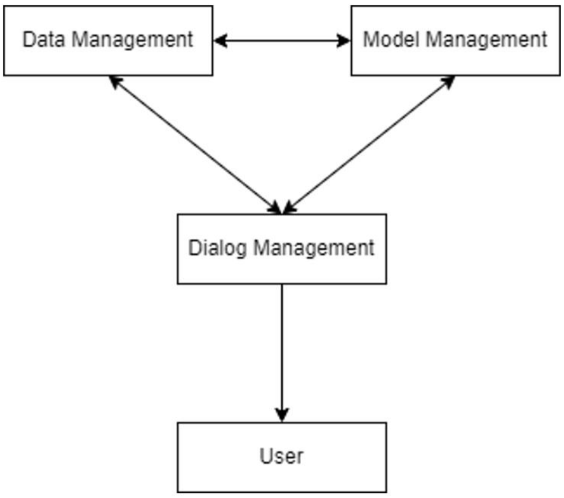


Fig. 1: Relationship between components

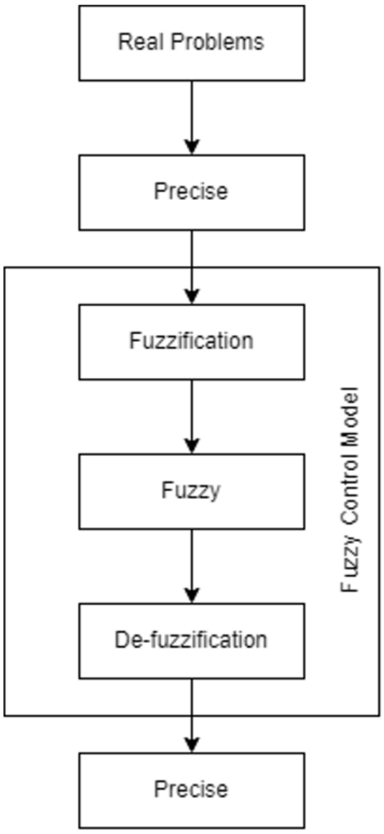


Fig. 2: Fuzzy logic algorithm (Utama, 2021)

Materials and Methodology

This research stage (Fig. 3) starts from the preliminary study stage, determines the parameters and data, designs and analyzes the model, constructs the model, and evaluates.

The first stage in creating a DSM is to undertake basic research to identify the primary difficulties. The early investigation suggested that identifying the best coder in the company is difficult. The purpose of choosing the best programmer is to determine which programmers will earn a raise, bonus, or promotion. Some businesses take the route of being chosen by the company's senior or management personnel. This method is viewed as unworkable and cannot be accepted equitably by all programmers.

After that, conduct interviews with business officials and investigate the criteria that will be used to choose the best programmer. Statistics may be gathered in a variety of ways, such as through surveys completed by programmers, assessments from superiors, or company-owned programmer's attendance data. Of course, the data is linked to the previously gathered parameters. The metrics and data collected during this method will be used as assessment criteria in the following step.

The model is constructed with a class diagram and a flowchart to show the model's stages or operations. The class diagram will describe three major classes: Personnel data, FL and SAW. The class criteria will describe the parameters that will be examined in the following stage. The flowchart diagram will show the process of processing raw data to obtain the outcome. Furthermore, the model development process is carried out by developing a website that corporate managers may use to enter the programmer's data and automatically obtain the calculation results. The model's output will be the value and ranking of each programmer in the firm.

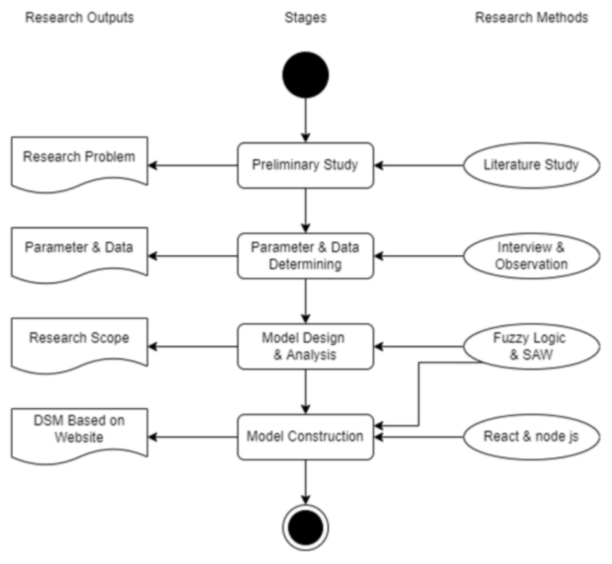


Fig. 3: Research stages

After the model has produced the best programming results, the verification and validation steps of the model must be completed. If the findings are deemed more accurate, the model developed will be utilized by the corporation to identify the top programmers in the future, as well as the best personnel in other divisions.

Results and Discussion

Parameterization

In this study, nine parameters were used to determine the best programmer in a company. Technical skills, problem-solving, communication skills, teamwork, discipline, work progress, time management, formal education, and informal education are all parameters that must work together in the created model. The constructed model will provide recommendations for the best programmers in the company based on the highest point.

The influence diagram (Fig. 4) shows the interconnected parts in the model and the methods (FL and SAW methods) to attain the aim of determining the top programmers in the firm. All parameters are determined using FL methods. To determine the boundaries of each parameter, the fuzzy triangular membership function with specified linguistic variable constraints is employed.

Each parameter has two linguistic categories namely bad and good. For technical skills, problem-solving, communication skills, teamwork, and discipline parameters (Fig. 5) have triangular bounds: (0, 0, 10, 40) and (10, 40, 51, 51). Furthermore, the work progress and time management parameters (Fig. 6) have triangle boundaries of (0, 0, 2, 8) and (2, 8, 10, 10). Finally, the parameters of formal education and informal education (Fig. 7) have triangular bounds of (0, 0, 0.5, 2.5) and (0.5, 2.5, 3, 3).

Each parameter will contain many sub-parameters that will be utilized as input values for the assessment to determine its value. The first parameter, technical skill, is divided into four sub-parameters: Knowledge of reacting, HTML, SQL, and MongoDB. The second parameter is communication skill, which is divided into three sub-parameters: Interpersonal communication, presentation, and language. The third parameter is problem-solving, which is broken down into three sub-parameters: Project, revision, and daily problems. The fourth parameter is teamwork, which is broken down into three sub-parameters: Responsibility, contribution, and project idea.

The fifth parameter is discipline, which contains three sub-parameters: On time, rules obeyed, and behavior. Sub-parameters of technical skill, problem-solving, communication skill, teamwork, and discipline (Fig. 8) will be classified as bad, enough, and good. With triangle limits of (0, 0, 10, 25), (12.5, 25, 37.5) and (25, 40, 50, 50).

The sixth parameter is a work progress, which has a single sub-parameter, the finished project. The seventh component is time management, which comprises sub-parameters such as project completion before the deadline. Work progress and time management sub-parameters (Fig. 9) will be categorized into three language categories: Bad, enough, and good. With triangle limits of (0, 0, 1, 5), (2.5, 5, 7.5) and (2.5, 5, 7.5), (5, 9, 10, 10).

The eighth parameter is formal education, which comprises a sub-parameter known as degree level. The last parameter is informal education, which contains a sub-parameter known as the number of certificates. Work progress and time management sub-parameters (Fig. 10) will be categorized into three linguistic categories: Bad, enough, and good. With triangle boundaries of (0, 0, 0.5, 1.5), (0.7, 1.5, 2.3) and (1.5, 2.5, 3.0, 3.0).

The Constructed Model

A class diagram will be used to describe the model in this research. A class diagram is a hypothetical modeling method that can be thought of as a high-level configuration diagram of object engagement. The class diagram in Fig. 11 depicts the relationship between the model's classes or entities.

Several classes are necessary to construct this DSM. The model is made up of five entities or classes: Programmer, FL, Membership Function, Fuzzy Rule, Triangular MF, Trapeziodal MF and SAW. The Programmer class specifies the characteristics that represent the evaluation of parameters of each programmer who works in the firm.

FL is used as a class and it contains two sorts of processes: Fuzzify and defuzzify. A Membership Function class and Fuzzy Rule are required to handle all actions in the FL class. All attributes in the Membership Function class describe the triangular membership function's attributes: Linguistic variable and degree of truth. The fuzzy Rule class describes the condition and rules of each parameter. Class Triangluar MF and Trapezoidal MF describe the limit attributes: Low bound to represent the lower limit value, mid Bound to represent the middle value, and up Bound to represent the upper limit value.

SAW is a class of methods that will be utilized when the fuzzy logic process has been completed. SAW may execute two sorts of processes: Measure Final Value and normalize Matrix. It requires some computations on the crisp value acquired in the fuzzy logic process to perform the two operations. The computation results from this class will be in the order of each programmer's rating in the firm. The results will be used to select the best programmer in the firm in the programmer class.

The model to be produced will be designed using nine parameters. Each parameter will be assigned a weight value chosen by the firm based on an agreement. Each parameter's weighting is calculated based on the most important parameters in determining the best programmer in the firm. The weight value will be multiplied by the value of each programmer in the SAW method. Table 1 shows the parameter.

Table 1: Parameter or criteria to determine the best programmer

No	Parameter	Weight
1	Technical skill	15
2	Problem solving	12
3	Communication skill	12
4	Teamwork	12
5	Discipline	8
6	Work progress	15
7	Time management	10
8	Formal education	8
9	Informal education	8

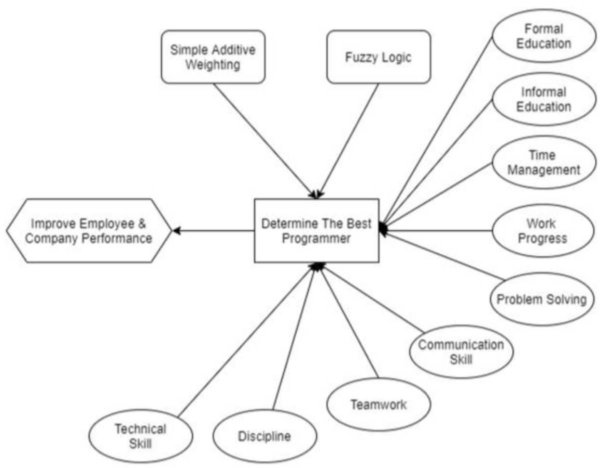


Fig. 4: Influence diagram to connect parameters, sub-parameter, and the constructed model objective

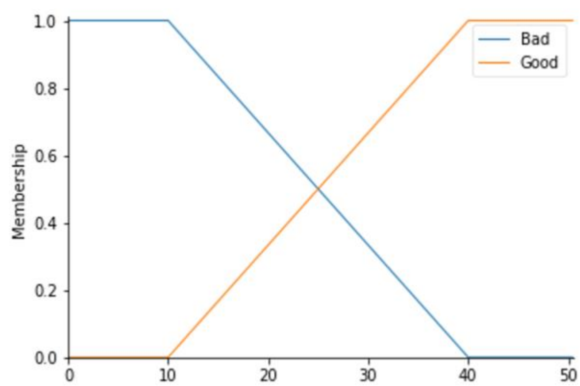


Fig. 5: The graph of fuzzy triangular membership function for parameter technical skill, problem-solving, etc.

DOI: 10.3844/jcssp.2022.530.539

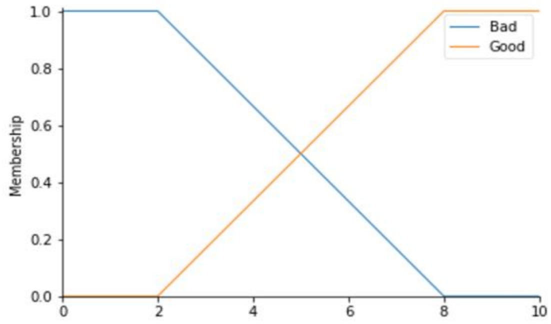


Fig. 6: The graph of fuzzy triangular membership function for parameter work progress and time management

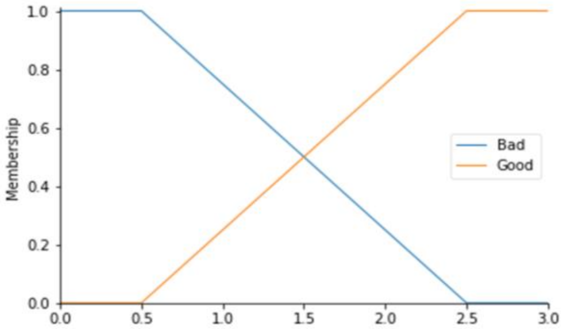


Fig. 7: The graph of fuzzy triangular membership function for parameter formal and informal education

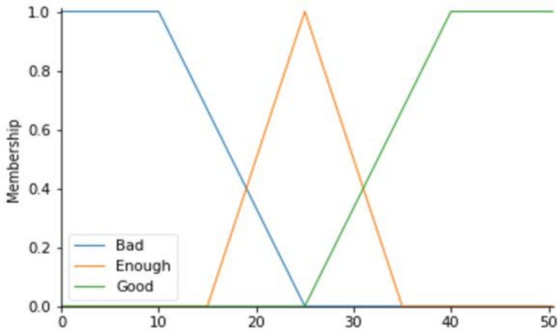


Fig. 8: The graph of fuzzy triangular membership function for sub-parameter react, etc.

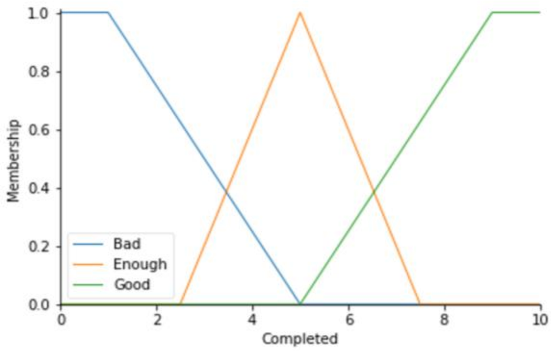


Fig. 9: The graph of fuzzy triangular membership function for a sub-parameter completed project, etc.

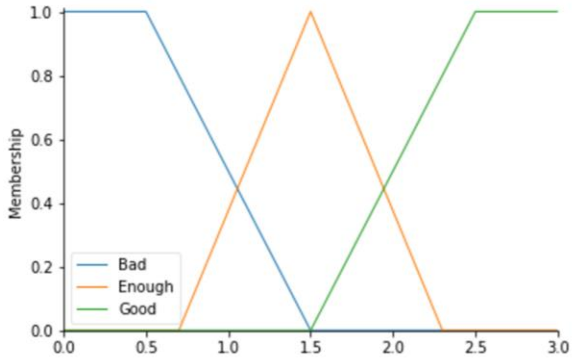


Fig. 10: The graph of fuzzy triangular membership function for a sub-parameter degree, etc.

The parameters of technical skill and work progress have the highest weight values. Because these two criteria are critical for a programmer. The technical skill parameter examines how many programming languages are mastered and used by the company's programmers. Meanwhile, the work progress parameter determines how many tasks the programmer finished throughout the appropriate period. Discipline, teamwork, communication skills, problem-solving, time management, formal education, and informal education are some of the other criteria. All these factors are also evaluated by the firm because they are essential standards that all programmers must fulfill.

The best programmer will be selected by the company's management. The decision will be made based on current data and will be made in front of all candidate of programmers. Each programmer will be assigned a value for each sub-parameter by the manager. Figure 12 and 13 provide examples of evaluations on various sub-parameters.

Following that, the value (Fig. 14) of the sub-parameters supplied by the company manager will be passed through the FL method (fuzzifying, defuzzifying) to obtain the crisp result. In addition to the SAW method, each programmer must do normalization calculations (Fig. 15). At this point, the benefit criteria computation is used if the weight of the parameter is increasing. The result for each programmer is calculated by multiplying the value of each parameter by a pre-determined weight (Fig. 16). In addition, the values of each parameter acquired by each programmer will be put together.

The results of the preceding computations (Fig. 17 and 18) will be utilized as a decision recommendation to choose the best programmer in the firm. The best programmer will be the one with the highest value from the Fuzzy Logic and Simple Additive Weighting calculations. Figures 17 and 18 show that Programmer 2 has the highest score with a total point of 97.94, while Programmer 8 has the lowest score with a total point of 91.17. Based on these total points, it can be determined that Programmer 2 is the best employee in the firm during this period, while Programmer 8 needs to improve his job performance.

DOI: 10.3844/jcssp.2022.530.539

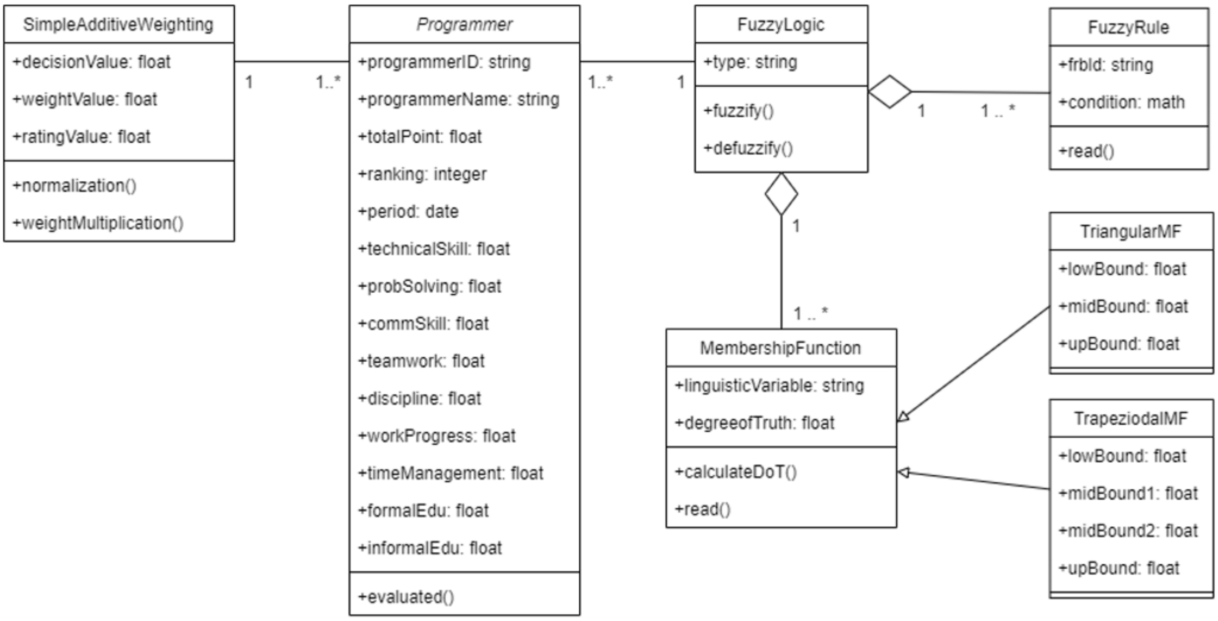


Fig. 11: The class diagram for the constructed model

Best Employee

Employee Point | Fuzzy Logic | Saw Normalization | Saw Weight | Final Point

Name	Technical Skill				Problem Solving			Communication Skill			Formal Education
	React	Html	Sql	Mongo	Project	Revision	Daily	With Others	Presentation	Language	Degree
Programmer 1	47	45	37	42	34	37	43	32	44	36	2
Programmer 2	29	47	45	50	37	35	45	30	47	27	3
Programmer 3	48	46	43	43	38	48	36	35	33	41	1
Programmer 4	39	42	47	29	40	35	35	44	40	36	3
Programmer 5	31	46	34	43	31	47	43	44	37	42	1
Programmer 6	50	42	38	38	47	38	30	31	29	37	2
Programmer 7	44	47	42	45	37	33	47	49	41	43	1
Programmer 8	32	42	41	50	36	46	40	43	38	46	1
Programmer 9	33	48	40	46	34	39	40	29	37	34	3
Programmer 10	39	46	28	38	49	49	35	48	43	41	2

Fig. 12: Constructed model dashboard (employee point)

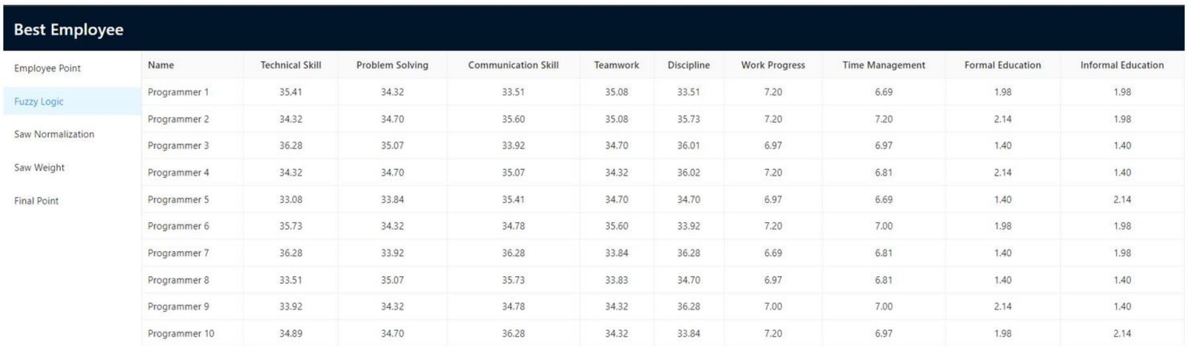
Best Employee

Employee Point | Fuzzy Logic | Saw Normalization | Saw Weight | Final Point

Name	Teamwork			Discipline			Work Progress	Time Management	Informal Education
	Responsibility	Contribution	Project Idea	On Time	Obey Rules	Behaviour	Completed	Before Deadline	Certificate
Programmer 1	45	31	24	31	30	32	10	7	2
Programmer 2	39	40	24	38	40	39	10	10	2
Programmer 3	35	34	25	44	39	42	8	8	1
Programmer 4	28	34	33	47	41	39	10	6	1
Programmer 5	36	47	35	35	44	35	8	7	3
Programmer 6	38	41	27	33	38	42	10	9	2
Programmer 7	31	45	31	49	44	45	7	6	2
Programmer 8	33	33	31	40	35	38	8	6	1
Programmer 9	40	34	39	49	46	45	9	9	1
Programmer 10	41	44	30	31	38	35	10	8	3

Fig. 13: Constructed model dashboard (employee point)

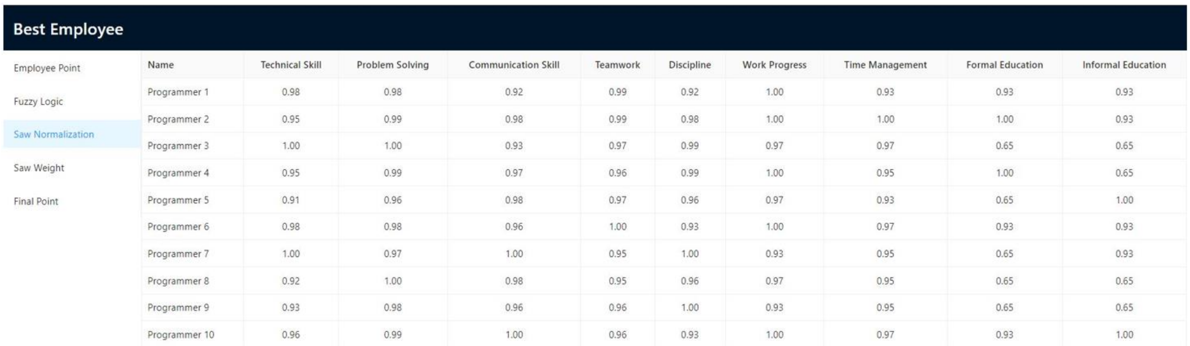
DOI: 10.3844/jcssp.2022.530.539


Best Employee

Employee Point
Fuzzy Logic
Saw Normalization
Saw Weight
Final Point

Name	Technical Skill	Problem Solving	Communication Skill	Teamwork	Discipline	Work Progress	Time Management	Formal Education	Informal Education
Programmer 1	35.41	34.32	33.51	35.08	33.51	7.20	6.69	1.98	1.98
Programmer 2	34.32	34.70	35.60	35.08	35.73	7.20	7.20	2.14	1.98
Programmer 3	36.28	35.07	33.92	34.70	36.01	6.97	6.97	1.40	1.40
Programmer 4	34.32	34.70	35.07	34.32	36.02	7.20	6.81	2.14	1.40
Programmer 5	33.08	33.84	35.41	34.70	34.70	6.97	6.69	1.40	2.14
Programmer 6	35.73	34.32	34.78	35.60	33.92	7.20	7.00	1.98	1.98
Programmer 7	36.28	33.92	36.28	33.84	36.28	6.69	6.81	1.40	1.98
Programmer 8	33.51	35.07	35.73	33.83	34.70	6.97	6.81	1.40	1.40
Programmer 9	33.92	34.32	34.78	34.32	36.28	7.00	7.00	2.14	1.40
Programmer 10	34.89	34.70	36.28	34.32	33.84	7.20	6.97	1.98	2.14

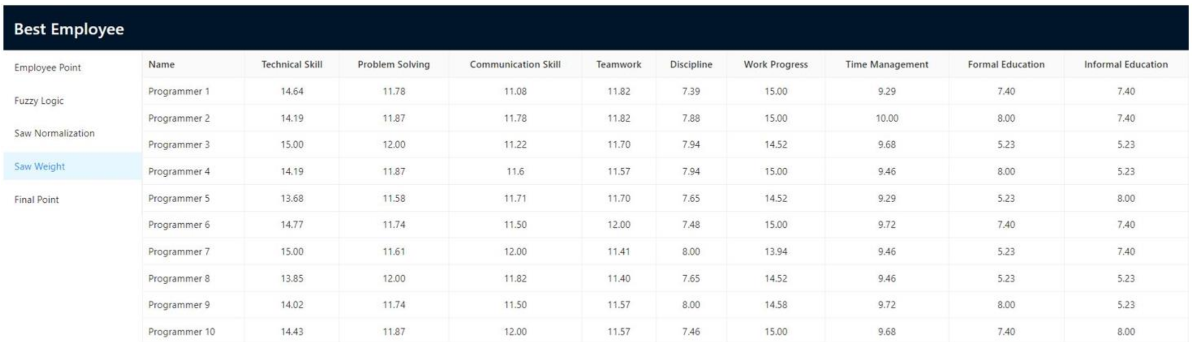
Fig. 14: Constructed model dashboard (the result of the fuzzy logic process)


Best Employee

Employee Point
Fuzzy Logic
Saw Normalization
Saw Weight
Final Point

Name	Technical Skill	Problem Solving	Communication Skill	Teamwork	Discipline	Work Progress	Time Management	Formal Education	Informal Education
Programmer 1	0.98	0.98	0.92	0.99	0.92	1.00	0.93	0.93	0.93
Programmer 2	0.95	0.99	0.98	0.99	0.98	1.00	1.00	1.00	0.93
Programmer 3	1.00	1.00	0.93	0.97	0.99	0.97	0.97	0.65	0.65
Programmer 4	0.95	0.99	0.97	0.96	0.99	1.00	0.95	1.00	0.65
Programmer 5	0.91	0.96	0.98	0.97	0.96	0.97	0.93	0.65	1.00
Programmer 6	0.98	0.98	0.96	1.00	0.93	1.00	0.97	0.93	0.93
Programmer 7	1.00	0.97	1.00	0.95	1.00	0.93	0.95	0.65	0.93
Programmer 8	0.92	1.00	0.98	0.95	0.96	0.97	0.95	0.65	0.65
Programmer 9	0.93	0.98	0.96	0.96	1.00	0.93	0.95	0.65	0.65
Programmer 10	0.96	0.99	1.00	0.96	0.93	1.00	0.97	0.93	1.00

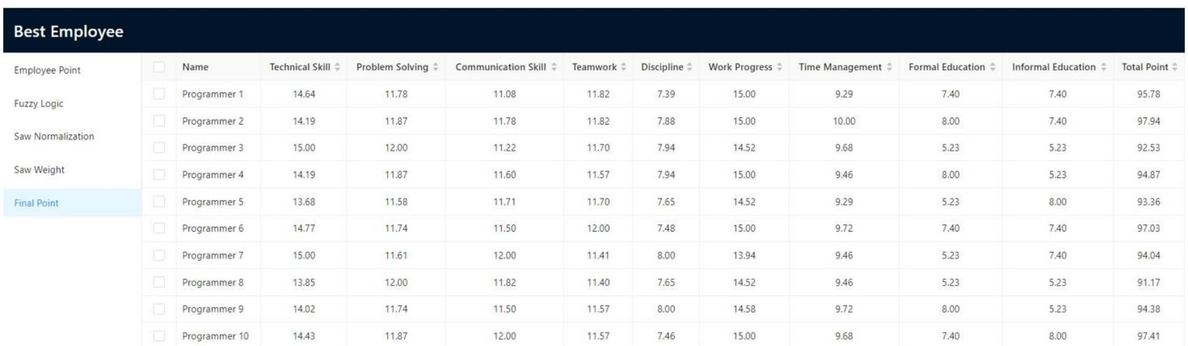
Fig. 15: Constructed model dashboard (the result of SAW normalization process)


Best Employee

Employee Point
Fuzzy Logic
Saw Normalization
Saw Weight
Final Point

Name	Technical Skill	Problem Solving	Communication Skill	Teamwork	Discipline	Work Progress	Time Management	Formal Education	Informal Education
Programmer 1	14.64	11.78	11.08	11.82	7.39	15.00	9.29	7.40	7.40
Programmer 2	14.19	11.87	11.78	11.82	7.88	15.00	10.00	8.00	7.40
Programmer 3	15.00	12.00	11.22	11.70	7.94	14.52	9.68	5.23	5.23
Programmer 4	14.19	11.87	11.6	11.57	7.94	15.00	9.46	8.00	5.23
Programmer 5	13.68	11.58	11.71	11.70	7.65	14.52	9.29	5.23	8.00
Programmer 6	14.77	11.74	11.50	12.00	7.48	15.00	9.72	7.40	7.40
Programmer 7	15.00	11.61	12.00	11.41	8.00	13.94	9.46	5.23	7.40
Programmer 8	13.85	12.00	11.82	11.40	7.65	14.52	9.46	5.23	5.23
Programmer 9	14.02	11.74	11.50	11.57	8.00	14.58	9.72	8.00	5.23
Programmer 10	14.43	11.87	12.00	11.57	7.46	15.00	9.68	7.40	8.00

Fig. 16: Constructed model dashboard (the result of SAW weight process)


Best Employee

Employee Point
Fuzzy Logic
Saw Normalization
Saw Weight
Final Point

Name	Technical Skill	Problem Solving	Communication Skill	Teamwork	Discipline	Work Progress	Time Management	Formal Education	Informal Education	Total Point
Programmer 1	14.64	11.78	11.08	11.82	7.39	15.00	9.29	7.40	7.40	95.78
Programmer 2	14.19	11.87	11.78	11.82	7.88	15.00	10.00	8.00	7.40	97.94
Programmer 3	15.00	12.00	11.22	11.70	7.94	14.52	9.68	5.23	5.23	92.53
Programmer 4	14.19	11.87	11.60	11.57	7.94	15.00	9.46	8.00	5.23	94.87
Programmer 5	13.68	11.58	11.71	11.70	7.65	14.52	9.29	5.23	8.00	93.36
Programmer 6	14.77	11.74	11.50	12.00	7.48	15.00	9.72	7.40	7.40	97.03
Programmer 7	15.00	11.61	12.00	11.41	8.00	13.94	9.46	5.23	7.40	94.04
Programmer 8	13.85	12.00	11.82	11.40	7.65	14.52	9.46	5.23	5.23	91.17
Programmer 9	14.02	11.74	11.50	11.57	8.00	14.58	9.72	8.00	5.23	94.38
Programmer 10	14.43	11.87	12.00	11.57	7.46	15.00	9.68	7.40	8.00	97.41

Fig. 17: Constructed model dashboard (total points of all parameters)

DOI: 10.3844/jcssp.2022.530.539

Fig. 18: Constructed model dashboard (total points of all parameters in graphical)

Conclusion

Technical skills, problem-solving, communication skills, teamwork, discipline, work progress, time management, formal education, and informal education are among the nine parameters examined in this study. Each of these parameters has its own set of sub-parameters. These sub-parameters will be assigned to each programmer as values.

The foundation of this method is fuzzy logic and a simple additive weighting method. The model is described in a class diagram using the object-oriented method and the results of these calculations and results are shown in the form of a website. The relationships between the entities or classes in each model are depicted in this figure.

There are ten programmers included in the constructed model. The company manager assigns a value to each programmer based on their performance while working for the firm. More study is required to achieve the best outcomes. You can add multiple parameters or criteria linked to programmer performance to acquire the best results in identifying the best programmer.

Acknowledgment

We would like to thank Bina Nusantara University for supporting and sponsoring our studies and work, especially the Bina Nusantara Graduate Program, Master of Computer Science.

Author’s Contributions

Lucky Christopher Chen: Analyzing all data, designing the model, and finalizing the manuscript.

Ditdit Nugeraha Utama: Reviewing and finalizing the model and manuscript.

Ethics

This manuscript substance is the author's original work and has not been previously published somewhere else. Authors already read and approved the manuscript and no potential ethical issues are immersed.

References

Anindita, M. (2020). Perbandingan metode profile matching dan simple additive weighting dalam sistem pendukung pengambilan keputusan penyeleksian anggota baru paduan suara cantus firmus. Yogyakarta: Universitas sanata dharma. https://repository.usd.ac.id/36864/2/155314067_full.pdf

Bashir, U., & Ramay, M. I. (2010). Impact of Stress on employees job performance a study on banking sector of Pakistan. International journal of marketing studies, 2(1), 122-126. https://ssrn.com/abstract=2281979

Daihani, D. U. (2001). Sistem pendukung keputusan. Penerbit elex media komputindo, Jakarta.

Faisol, A., Muslim, M. A., & Suyono, H. (2014). Komparasi Fuzzy AHP dengan AHP pada sistem pendukung keputusan investasi properti. Jurnal EECCIS, 8(2), 123-128. https://jurnaleeccis.ub.ac.id/index.php/eeccis/article/view/249

Fitriana, J., Ripanti, E. F., & Turisna. (2018). Sistem pendukung keputusan pemilihan mahasiswa berprestasi dengan metode profile matching. Jurnal sistem dan teknologi informasi, 153-160. doi.org/10.26418/justin.v6i4.27113

Manulang, M. (2002). Manajemen personalia, Ghalia Indonesia, Jakarta.

Nardiono, N. (2017). Komparasi metode Simple Additive Weightin (SAW) dan metode Weighted Product (WP) dalam menentukan karyawan terbaik (Studi Kasus: PT. matrixnet global Indonesia). Jurnal Informatika Universitas Pamulang, 2(1), 25-33.

Sahir, S. H., Rosmawati, R., & Minan, K. (2017). Simple additive weighting method to determining employee salary increase rate. International journal of scientific research in science, engineering and technology, 3(8), 42-48. https://www.researchgate.net/publication/348161939_IJSRST1173821

Scott Morton, M. S. (1970). Program management and interactive management decision systems. https://dspace.mit.edu/bitstream/handle/1721.1/48798/programmanagemen00scot.pdf?sequence=1

Susilo, A. A. (2017). Penerapan metode profile matching pada sistem pendukung keputusan pemilihan ketua program study. JUITA jurnal informatika, 5(2), 87-93. doi.org/10.30595/juita.v5i2.1939

Turban, E., Sharda, R., & Delen, D. (2011). Decision support and business intelligence system. New Jersey, United State of America: Prentice Hall.

Utama, D. N. (2021). Logika fuzzy untuk model penunjang keputusan. Garudhawaca.

Utama, D. N., & Oktafiani, S. (2020). Generic decision support model for determining the best marketer. Advances in science, technology and engineering systems journal, 5(4), 841-848. doi.org/10.25046/aj050498

Utama, D. N., & Rustamaji, E. (2018). Fuzzy decision support model for human resources performance appraisal. 2018 International conference on information management and technology (ICIMTech), 272-276. doi.org/10.1109/icimtech.2018.8528145

Vosloban, R. I. (2012). The influence of the employee's performance on the company's growth-A managerial perspective. Sci Verse science direct, 3, 660-665. doi.org/10.1016/s2212-5671(12)00211-0

Yaqin, A., Utami, E., & Luthfi, E. T. (2014). Sistem pendukung keputusan penentuan dosen pembimbing dengan metode logika fuzzy. Seminar nasional informatika, 1(1), 146-159. http://e-journal.potensi-utama.ac.id/ojs/index.php/SNIf/article/view/223

Yuliani, I. D. (2013). Sistem pendukung keputusan pemilihan karyawan terbaik dengan metode analytical hierarchy process. Journal sisfotenika, 3(2), 105-114. doi.org/10.30700/jst.v3i2.53

Yusuf, L., Hidayatulloh, T., Nurlaela, D., Utami, L. D., & Hasan, F. N. (2020). Simple additive weighting untuk front-end framework terbaik. ILKOM Jurnal ilmiah, 12(2), 136-142. doi.org/10.33096/ilkom.v12i2.593.136-142

Zadeh, L. A. (1965). Fuzzy sets. Information and control, 8(3), 338-353. doi.org/10.1016/S0019-9958(65)90241-X

Zadeh, L. A. (1975). The concept of a linguistic variable and its application to approximate reasoning. Information sciences, 8(3), 199-249. doi.org/10.1016/0020-0255(75)90036-5