Application of C 4.5 Algorithm Method for Determining the "*Bidik Misi*" Scholarship Admission in Institute Of Informatics and Business (IIB) Darmajaya

Halimah[1], Deppi Linda[2], TM Zaini[3], Agus Rahardi[4]

Faculty of Computer Science, Informatics and Business Institute Darmajaya,
Jl. Z.A. Pagar Alam, Bandar Lampung 35142, Indonesia
E-mail: halimahyunus@darmajaya.ac.id, deppilinda@darmajaya.ac.id, tmzaini@darmajaya.ac.id, agus.rahardi@darmajaya.ac.id

Abstract: The Bidik Misi Scholarship is one of the educational scholarships given by the Government of the Republic of Indonesia to prospective students who have high academic potential with underprivileged conditions. IIB Darmajaya still uses the manual method to determine students who are eligible for scholarships so that they do not meet the target and take a long time. This Research implementation of C 4.5 algorithm to determine prospective students who are entitled to receive the Bidik Misi Scholarship from IIB Darmajaya using the Rational Unified Process (RUP) method. The tools used in this research are Use Case Diagrams, Activity Diagrams, Sequence Diagrams, and Class Diagrams. The programming languages used in this research are PHP and MySQL. The results of this study indicate that this system is not determined only to determine prospective students who are entitled to receive the Mission Scholarship, but is also able to make it easier for the committee to work more quickly and efficiently.

Keywords: *"Bidik Misi" Scholarship, Rational Unified Process (RUP), C 4.5 Algorithm"*

1. INTRODUCTION

Bidikmisi Scholarship was the tuition aid from the Government of the Republic of Indonesia through the Directorate General of Higher Education, Ministry of Education and Culture, for prospective students who had disadvantages economically and had potentials academically at higher education until graduation on time. This program started since 2010.

This tuition aid provided since the prospective students admitted to higher education for 8 semesters in Diploma IV and Undergraduate program and for 6 semesters for Diploma III program. This scholarship was in the form of exemption from all tuition fees during in higher education. In addition, the scholarship recipients also received the academic fees every 6 months. The target of the *Bidikmisi* program was the graduates of Senior High Schools / Vocational Schools / Islamic Senior High Schools / Islamic Vocational Schools or the other equivalent education units who had economically disadvantages and had good academic potentials.

The number of recipients of *Bidikmisi* scholarships at IIB Darmajaya from 2015 was 25 people; 2016 was 48 people; 2018 was 27 people. The number of recipients who did not receive this scholarship was various depending on the portion determined from the Higher Education. Therefore, there were some of them who accepted because they met the criteria to receive the scholarship. The number of participants applying for *Bidikmisi* scholarships was large. The indicators in selecting the scholarship application files still used manual methods – the students who were eligible for scholarships took the scholarship – so that it also took a long time and was often not on target.

2. LITERATURE REVIEW

The decision support system (DSS) was one of the software products in the form of the result of solving the problems itself. It related to the answers to questions about 'what to do and so on. The decision made to deal with problems or mistakes that occurred against the plans. The level of decision-making tasks was equivalent to the planning tasks in the organization. [1].
The decision tree algorithm based on a divide-and-conquer approach to classifying a problem. The algorithm worked from top to bottom looking at each stage of the attribute to divide it into the best part of the class and recursively processing the resulting sub problems from the division. This strategy produced a decision tree that converted into a set of classification rules [2].

In the decision tree, there were three types of nodes e.g.:
    a. Root Node is the top node. There no input and had no output or had more than one output.
    b. Internal Node was the branching node. They have only one input and had at least two outputs.
    c. Leaf node or terminal node is the end node. There is only one input and no output.

As shown in Figure 1, the decision tree depended on if-then rules but did not require parameters and metrics. The simple and interpretable structure allowed the decision tree to solve multi-type attribute problems. Decision tree also managed missing values or noise data [4].
There were many algorithms used in the formation of a Decision Tree, including ID3, CART, and C4.5 [6]. The C4.5 algorithm and the decision tree were two inseparable models because of the developed decision tree. The C4.5 algorithm was needed [5].
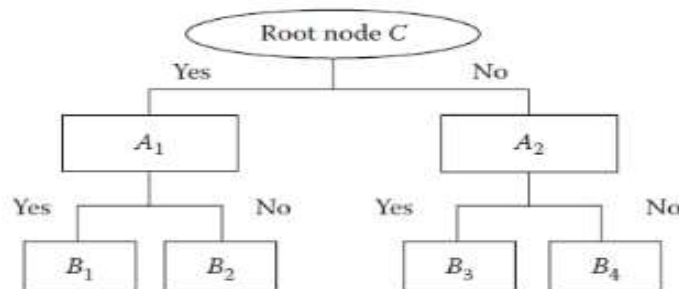

Figure 1. Example structure of Decision Tree
Source: [6]

The classification with a Decision Tree using the C4.5 Algorithm through several stages was as follows [6]:
    a. Preparing the training data usually taken from the historical data or past data, which was made into certain classes.
    b. Calculating the entropy value used to calculate the gain value of each attribute so that the attribute with the highest gain value obtained and used as the root of the tree.
    c. Repeating the previous step by calculate the value of each attribute based on the highest gain value until all records were partitioned.
    d. The process of this Decision Tree stopped if all the records in node N were assigned in the same class. There were no attributes in the partitioned records and no records in the empty branch.

3. METHOD

Developers and customers met and defined the overall object of the software, identified all known needs, and did "flash design. ". Design flash focused on the presentation of those aspects of the software that was visible to the customers or users (e.g., input approaches and output formats). Quick design led to the construction of a prototype [7] . The prototype was evaluated by the customer and used to filter software development needs, can seen in figure 2.
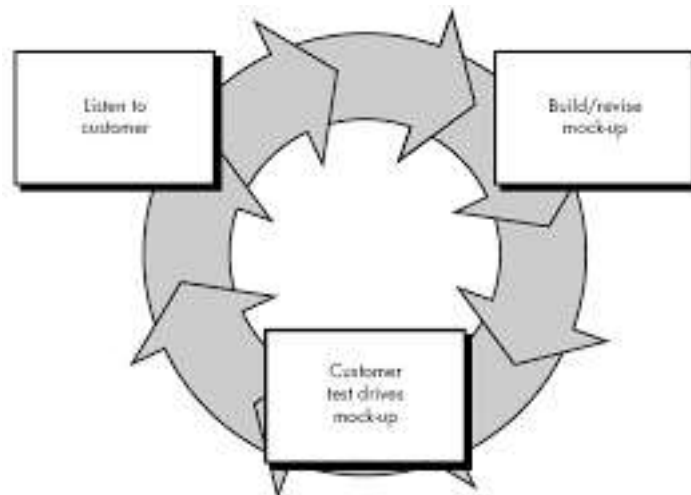

Figure 2. Prototype Paradigm

Prototype model was also defined as the process of developing a prototype quickly used at first and continuously improved until a complete system was obtained. Prototype modeling was a process used to assist software developers in forming prototypes of software that must be made. The process in the prototyping model was explained as follows:

    a. Requirements gathering: Developer and client met and defined general goals, known needs, and a description of the parts.

    b. Design: The design was carried out quickly. The design represents all known aspects of the software. The design became the basis for making the prototype.

    c. Prototype evaluation: The client evaluated the prototype made and used to clarify software requirements.

The repetition of these three processes continued until all needs were met. Prototypes were made to satisfy client needs and to build software faster but not all prototypes were used. For client needs, it was better if the prototypes that were made were utilized.

This research used the data collecting techniques so that it complemented the existing data. The stages of the method used were as follows:

    a. Interview: The author conducted interviews with several committees in determining the eligibility of *BidikMisi* scholarship recipients.

    b. Observation: The author obtained data and facts from direct observation at the research location.

    c. Literature review: conducting research by studying books and literature in libraries, academies, e-journals or from other places directly or indirectly.

a. Systems Development Method

The system development method in this final report used the Rational Unified Process (RUP) method. This RUP used the concept of object oriented that focused on developing models using the Unified Modeling Language (UML) [8].

In software development, the author used the first dimension RUP depicted horizontally. The first dimension of the RUP represented the dynamic aspects of software development [9].

b. Application of Methods Rational UnifiedProcess

The first dimension in the Rational Unified Process consisted of several phases e.g., Inception, Elaboration Construction, and Transition. These phases were carried out to develop software e.g.:

1) Inception

The stages carried out in this phase included:

a) Determining the website coverage for determining *BidikMisi* scholarship recipients.
b) Determining the requirements / needs analysis which consisted of:
    -   HR needs
    -   Functional requirements
    -   Hardware requirements
    -   Software requirement

2) Elaboration

The stages carried out in thorugh several phases, e.g.:

a) Making use-case diagram modeling
b) Creating a narrative use-case
c) Creating an activity diagram
d) Creating a sequence diagram
e) Creating a database design
f) Creating a user interface design

3) Construction

The stages carried out at this stage were in the form of implementation. At this stage, the implementation of the software user interface design was carried out.

4) Transition

Transition was the phase for the transition period during implementing the system in the user environment. This phase began with a beta release.

In this phase, the overall system development had been complete and the software will be hand over to the users to start the flow of its usage. At the end of this phase, the decision made whether the product released or not based on the level of user satisfaction. If there was a shortage, another cycle initiated to increase the system capability.

4. RESULT AND DISCUSSION

a. Use Case Diagram

Use case diagrams described the expected functionality of a system that described the overall work of the system in outline by presenting the interactions between the actors made and providing an overview of the functions of the system. Figure 3 can see in system use case diagram.
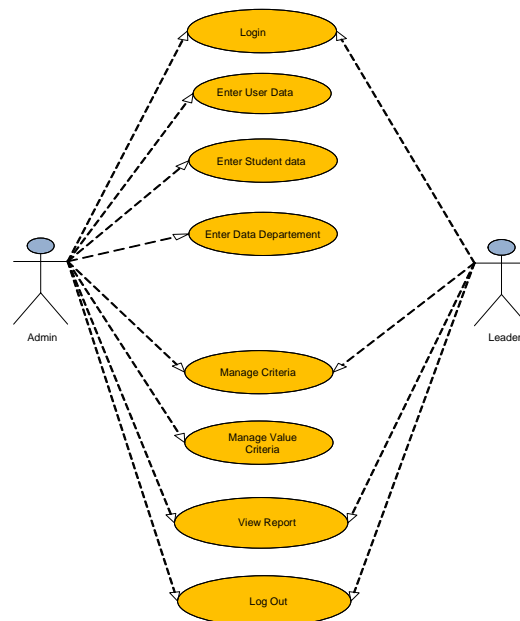
Figure 3. Use Case Diagram

b.   Implemention

1) Start Page View
Before entering the website, prospective students must log in. The student login page required a username and password so that students must register first. After students loggen in, they went to the home page of the website. Start page view can see in figure 4.
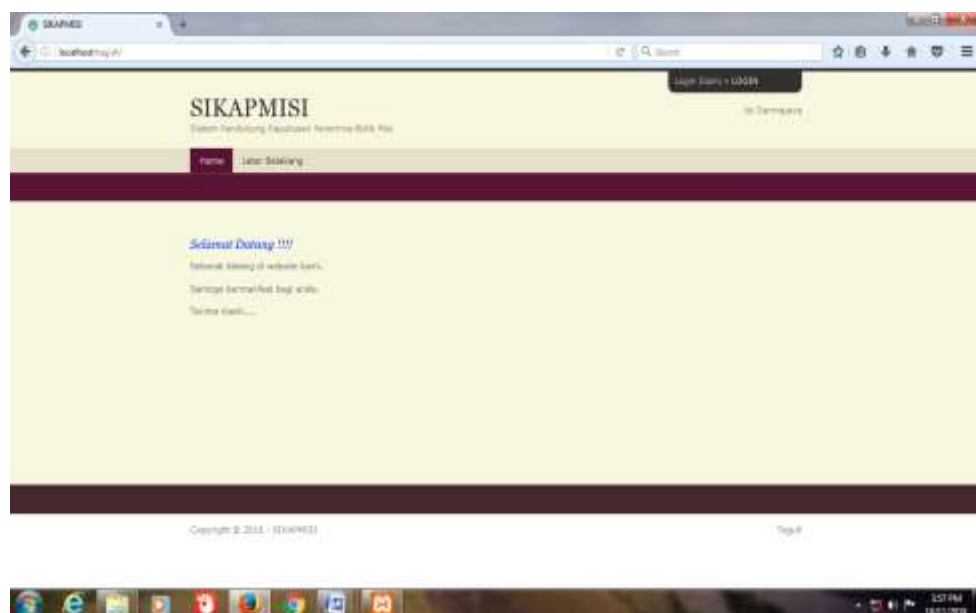

Figure 4. Start page view

2) Login Page

Before students enter the website, a menu will appear as shown in figure 5. After students logged in, they went to the home page of the website.
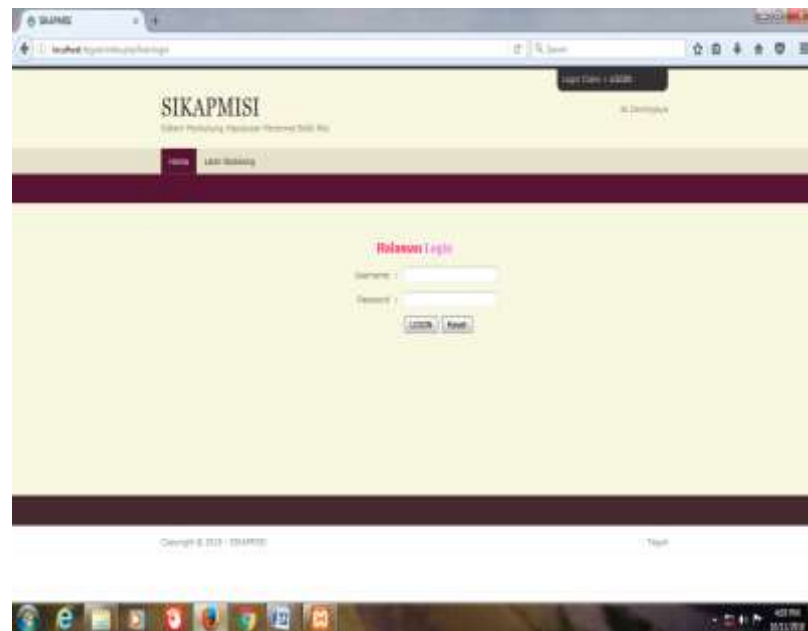

Figure 5. Login page display

3) Admin Page

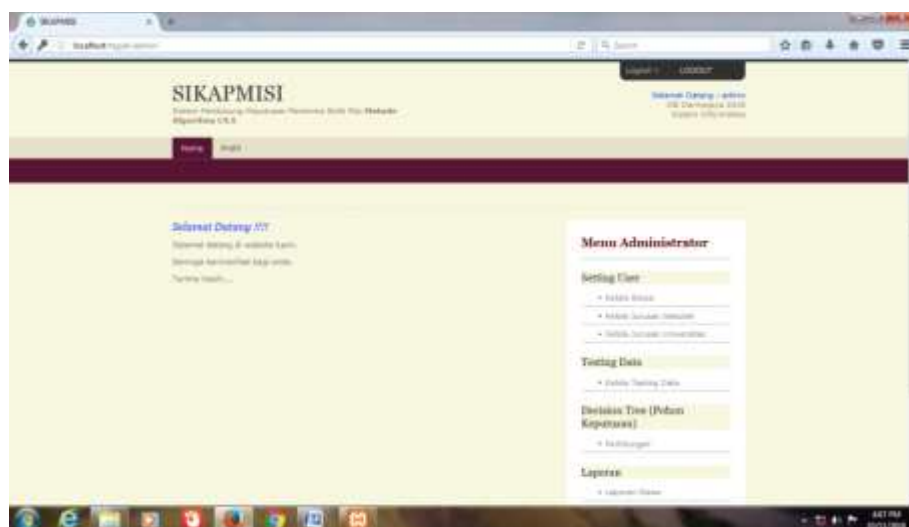The following was a display of the admin menu as a system manager. Admin page can see in figure 6.

Figure 6. Admin Menu Display

4) Student Management Page

The student management menu was the administrator's menu display for managing the data of prospective *BidikMisi* scholarship recipients. Menu display students' management page as show in the figure 7.

Figure7. Menu Display Manage Students

5) Menu Display Manage School Departments

The following administrator view was to manage the existing department data from each prospective school for *BidikMisi* scholarship recipients

Figure 8. Menu Display Manage School Departments

6) Faculty Management Page

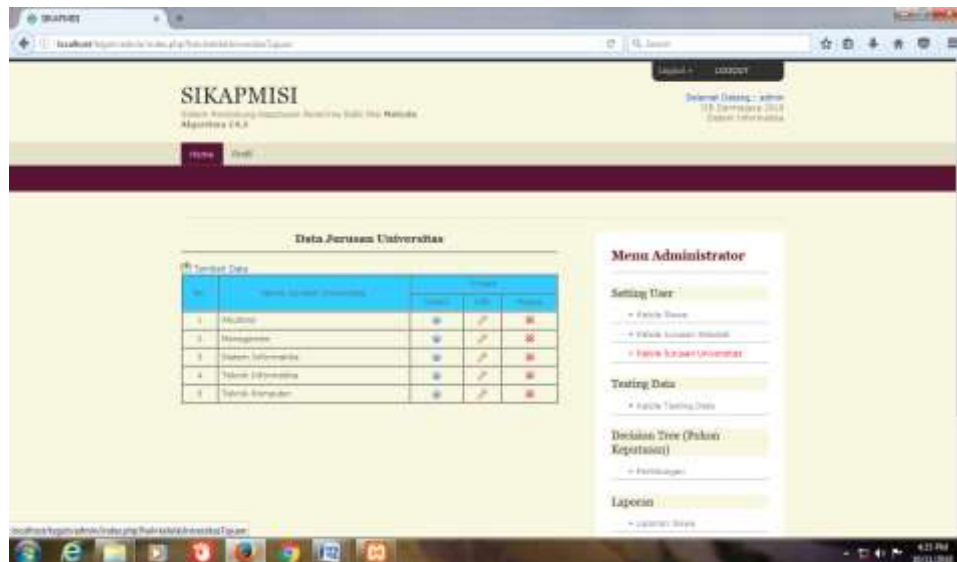Adminstrator menu displayed to list all majors in the institution. Faculty management page can see in figure 9.

Figure 9. Menu Display Manage University Departments

7) Menu Display Manage Testing Data

Display Menu Manage Testing Data for administrators to manage data on prospective scholarship recipients who deserved a *BidikMisi* scholarship

Figure 10. Menu Display Manage Testing Data

8) Calculation Menu Display

The following image showed the calculation menu using the C.45 algorithm to manage the incoming data

Figure 11. Calculation Menu Display

9) Report Menu Display

The following picture was a menu display of the report of students who received *BidikMisi* scholarships.

Figure 12. Report Menu Display

c. System Integration and Testing
    It was focused on testing of the software functionally and ensuring all parts that had been tested. The correctness of the tested software was only seen based on the output generated from the data or input conditions given to existing functions. From the resulting output, the program's ability to meet user needs were measured at the same time to identify errors and ensured that the resulting output was needed.

d. Operation and Maintenance
    Software maintenance experienced changes while it was communicated to customers (a possible exception was embedded software). Changes occurred because the errors were determined and the software had to be adapted to accommodate changes in the external environment (for example, changes were needed as a result of a new peripheral device or operating system) or because the customers needed functional or performance developments. Software maintenance applied each of the previous program phases again and did not create new ones.

e. The Work Process of Implementing the C4.5 Algorithm in the Selection of *BidikMisi* Scholarship Recipients
    The process of selecting recipients of *BidikMisi* talents was only the input by the administrators. The administrators must login to enter the admin menu. After logging in, the administrators entered the student file data into the student management menu. The administrators inputted the results of the criteria or selection values into the data testing management menu. After managing data testing, students were automatically selected by the system. The result of this testing was accepted or rejected was seen in the mission report menu.

5. CONCLUSION

The application of the C4.5 Algorithm Method in Determining *BidikMisi* Scholarship Recipients at IIB DARMAJAYA was that:
a. The applications were made through the C4.5 Algorithm Method in Determining *BidikMisi* Scholarship Recipients.
b. This application was intended for those who wanted to determine the eligibility of prospective *BidikMisi* scholarship recipients so that it was faster and more efficient. The prospective students also found out whether their requirements were approved or not by the relevant committee. The application was also used for administrators or operators to facilitate their work in determining the eligibility of prospective students who received *BidikMisi* scholarships and prospective student reports.

REFERENCES

[1] Musthafa, Aziz, Hadi Suyono, and Moechammad Sarosa. "Perbandingan Kinerja Algoritma C. 45 dan AHP-TOPSIS Sebagai Pendukung Keputusan Proses Seleksi Penerima Beasiswa." *Jurnal EECCIS* 9.2 (2016): 117-122.

[2] Salafudin, M. (2020). *SISTEM PENDUKUNG KEPUTUSAN UNTUK PENERIMAAN BEASISWA BAGI SISWA KURANG MAMPU MENGGUNAKAN METODE SIMPLE ADDITIVE WEIGHTING (SAW)(Studi Kasus: Sekolah Dasar Pada Gugus Hos Tjokroaminoto Kecamatan Sukorejo)* (Doctoral dissertation, University of Technology Yogyakarta).

[3]   Syafar, A.M. (2018). Sistem Pengambilan Keputusan Memilih Program Studi di UIN Alaudin

[4]   Han, J., & Kamber, M. 2006. Data Mining Concept And Tehniques. San Fransisco: Morgan Kauffman. ISBN 13: 978-1-55860-901-3

[5]   SUTEDI, S., &Agarina, M. (2017). Implementasi Rational Unified Process Dalam Rancang Bangun Sistem Informasi Penjualan Hasil Bumi Berbasis Web Pada CV. Aneka Mandiri Lestari Bandar Lampung. *Explore: Jurnal sistem informasi Dan Telematika (Telekomunikasi, Multimedia Dan Informatika)*, *8*(2).

[6]   Yunus, H., Linda, D., &Klaralia, F. (2020). Penerapan Algoritma Naïve Bayes Untuk Memprediksi Penyakit Malaria Pada Puskesmas Hanura. *TEKNIKA*, *14*(1), 57-63.

[7]   Herawati, Y. (2020). *SISTEM PENDUKUNG KEPUTUSAN REKRUTMEN TENAGA KERJA INDONESIA PADA PT. SUKSES MANDIRI UTAMA MENGGUNAKAN METODE NAÏVE BAYES BERBASIS WEB MOBILE* (Doctoral Dissertation, IIB DARMAJAYA).

[8]   Halimah, H., &Anggi, A. (2019). Pengembangan Sistem Informasi Penelitian Dan Pengabdian Dosen Pada Jurusan Di IBI DARMAJAYA Berbasis Web. *TEKNIKA*, *13*(1), 11-17.

[9]   Musthafa, Aziz, Hadi Suyono, and Moechammad Sarosa. "Perbandingan Kinerja Algoritma C. 45 dan AHP-TOPSIS Sebagai Pendukung Keputusan Proses Seleksi Penerima Beasiswa." *Jurnal EECCIS* 9.2 (2016): 117-122.