Journal of Information Systems and Informatics
Vol. 5, No. 4, December 2023 e-ISSN: 2656-4882 p-ISSN: 2656-5935
DOI: 10.51519/journalisi.v5i4.588 Published By DRPM-UBD

The Development of Web-based Application for Private School Tuition Fee Management with Prototyping Model

Jansen Wiratama[1], Monika Evelin Johan[2], Sobiyanto[3], Matthew Chandra[4], Victor Ilyas Sugara[5]

[1,2]Information Systems, Universitas Multimedia Nusantara, Tangerang, Banten, Indonesia.
[3,4]Informatics, STMIK Dharma Putra, Tangerang, Banten, Indonesia.
[5]Computer Science, Universitas Pakuan, Bogor, Indonesia.
Email: [1]jansen.wiratama@umn.ac.id, [2]monika.evelin@umn.ac.id, [3]sobi_yanto@yahoo.co.id, [4]mattchndr@gmail.com, [5]victor.ilyas@unpak.ac.id

Abstract

Private schools need help in handling school fees and financial processes. Traditional manual payment systems result in data processing issues, delayed financial reporting, and complications from misplaced records. Late fee payments threaten school income, which is crucial for staff salaries. Modern solutions are imperative. Desktop applications have limitations, requiring installation on specific devices, leading to compatibility concerns. This research opts for a web-based application. It employs prototyping models and predictive abilities using the Naïve Bayes algorithm. The web-based application aims to streamline fee management and predict payment delays, enhancing financial transaction management while prioritizing data security through database encryption. This web-based solution aligns with private schools' operational needs, simplifying payments and increasing late payment prediction accuracy. Extensive black-box testing validated its suitability, satisfying administrative staff needs. Four test cases gained administrative team approval. This innovation empowers private schools to optimize operations and financial management. In summary, the research tackles critical financial challenges private schools face by introducing a web-based application that simplifies payment processes, enhances accuracy in predicting late payments, and aligns seamlessly with administrative needs.

Keywords: Application, Naïve bayes, Private School, Prototyping, Web-based.

1. INTRODUCTION

Education in Indonesia continues to experience development and progress every year. In the era of globalization and information technology, which is increasingly developing rapidly, information and communication technology have become an urgent need in education [1]. Technology influences learning and supports school administration and finances [2]. A critical component in school administration is the Educational Development Contribution or tuition fee, a financial contribution paid by school students. tuition fee is vital in financing education provision so that teaching and learning activities can occur well. With tuition fee, schools can obtain additional income apart from funding sources such as School Operational

Assistance Funds or Bantuan Operasional Sekolah (BOS) and educational donations [3].

One of the private elementary schools in Indonesia, which is the object of this research, is a private school located in Pasar Kemis, Tangerang. In the business process of paying tuition fees, this school still uses conventional methods, which gives rise to various problems related to paying tuition fees. The problems faced by this school including the tuition payment process, which is still conventional on recording tuition payment data process, which still relies on physical ledgers. This results in problems in data processing, delays in preparing financial reports, and complexity if files or payment receipts are lost or damaged. Apart from that, late payment of tuition fees by students is also a severe problem because it can reduce school income, which is used to pay the salaries of teachers and employees.

A more modern and efficient solution is needed to improve tuition fee management. One solution is to implement the prototyping model to develop web-based applications and add the features of prediction using the Naïve Bayes algorithm. The goals are to build a web-based application to manage tuition payments and students' prediction delays of tuition fee date payments [4]. With the prototyping model, the steps in designing and developing an application will help this school to manage financial transactions and predict students who may be late paying tuition fees. This system will also optimize data security by implementing database encryption [5]. The implementation of this web-based application can provide a solution for a School in overcoming their tuition payment problems, enabling the system to be more in line with the school's needs and producing efficient and up-to-date solutions to simplify the process of operational activities at the school in payment as well as predicting late tuition payment payments [6].

This prototyping model is instrumental in developing web-based applications, allowing developers to create an initial version or "mock-up" of a website before making a complete final version. This way, they can test initial ideas, design user interfaces, and better identify requirements before committing to complete development. Prototyping helps minimize the risk of errors and ensures that the website will meet user needs well [7]. The intention of the implementation of the Naive Bayes Algorithm is to predict student payment delays. Integrating these algorithms within the web-based application allows websites to become more reliable to users, such as school administrators, so that they can make better decisions [9].

2. METHODS

The use of prototyping models in software development has become an increasingly important approach in ensuring the success of information

technology projects [10]. This model allows developers to design, test, and understand user needs better before creating the final version of the software [11].

These prototyping models serve as a tool to visualize concepts, design user interfaces, and identify potential problems early on. Using this model can save time and resources by allowing developers to correct and improve the design and functionality of the software before it reaches the implementation stage [12].

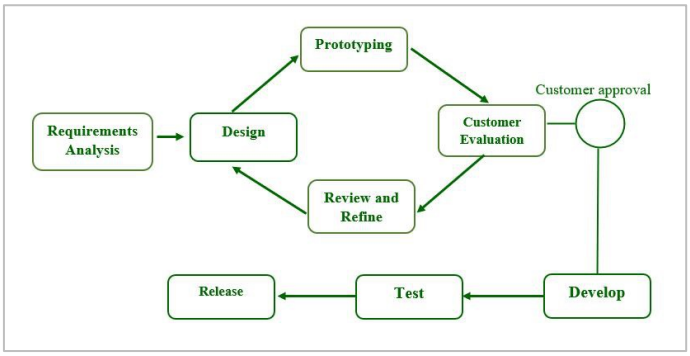

Figure 1. Prototyping Model Phases: six SDLC phases

Figure 1 depicts the stages of requirements engineering in the prototyping model. A good understanding of the steps to be taken in a development method is essential [13]. The following are the stages in the prototyping model:

1. **Stage 1-Requirement Gathering and Analysis:** This stage is the initial step in implementing the prototype model on web-based application development. In this stage, users are asked what they expect or want from the system.
2. **Stage 2-Quick Design:** The initial stage of prototype web-based application development is the utilization of the Unified Modeling Language (UML), a widely recognized modeling language. The utilization of UML has been employed as a method for visually depicting, generating, and documenting web-based applications and architectural designs [14].
3. **Stage 3-Build a Prototype:** This stage will culminate in the creation of a prototype application. The mock-up of application will be created utilizing design tools such as FIGMA. In this prototype, users are afforded the ability to apprehend the advancement of the software visually.
4. **Stage 4-Initial User Evaluation:** This stage will evaluate web-based applications. During this stage, users will actively participate in the process of functionality testing, and after that, the users will offer their feedback.

5. **Stage 5-Refining Prototype:** After obtaining user feedback from the initial user evaluation, system developers use the prototyping model to create a more complete and detailed system. If the user gives any input, the final design is approved by improving the client's response to feedback and suggestions.
6. **Stage 6-Implement Product and Maintain:** The final stage of implementing this Prototyping system development model is implementing a web-based application on the research object and carrying out maintenance according to the needs and schedule that have been determined.

Apart from using a prototyping model to develop the web-based application, this research also implements the Naive Bayes Algorithm to make predictions on tuition fee payment delay based on existing data that has been provided. The Naïve Bayes Algorithm was chosen because it is known to have a high level of accuracy with simple calculations [15]. The equation of Bayes' theorem is shown in Equation 1.

$$P(H/X)=\frac{P(X/H).P(H)}{P(X)} \quad (1)$$

Descriptions:

X : Data with unknown class
H : Data hypothesis X is a specific class
P(X|H) : Probability of hypothesis H based on condition X (posterior probability)
P(H) : Probability of hypothesis H (prior probability)
P(X|H) : Probability of X based on these conditions.
P(X) : Probability of X

Based on the explanation above, P(H|X) describes how big a chance H will occur when we have information X, such as the percentage of H in is similar to the rate of presence of X in H. Furthermore, P(H) is the initial probability that H will occur without considering X to the possibility of H occurring and vice versa.

In the context of evaluating the web-based application according to the user needs and able to provide more accurate predictions or analysis, User Acceptance Testing (UAT) with a black box testing model is carried out by the users [16]. The black box testing model ensures the web-based main functions run well, meet user needs and expectations, and deliver the expected results [17]. User Acceptance Testing (UAT) also helps ensure that the web-based model has been thoroughly tested by actual end users, thereby increasing the web-based chances of success in the market and reducing the risk of problems arising after launch [18].

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

3. RESULTS AND DISCUSSION

Based on the results of implementing the Prototyping model stage 1-Requirement Gathering and Analysis, which was carried out through interviews with resource persons, namely administrative staff and school principals, the results of the analysis of system requirements for the research object were obtained. The needs analysis is depicted according to the flowchart in Figure 2.:

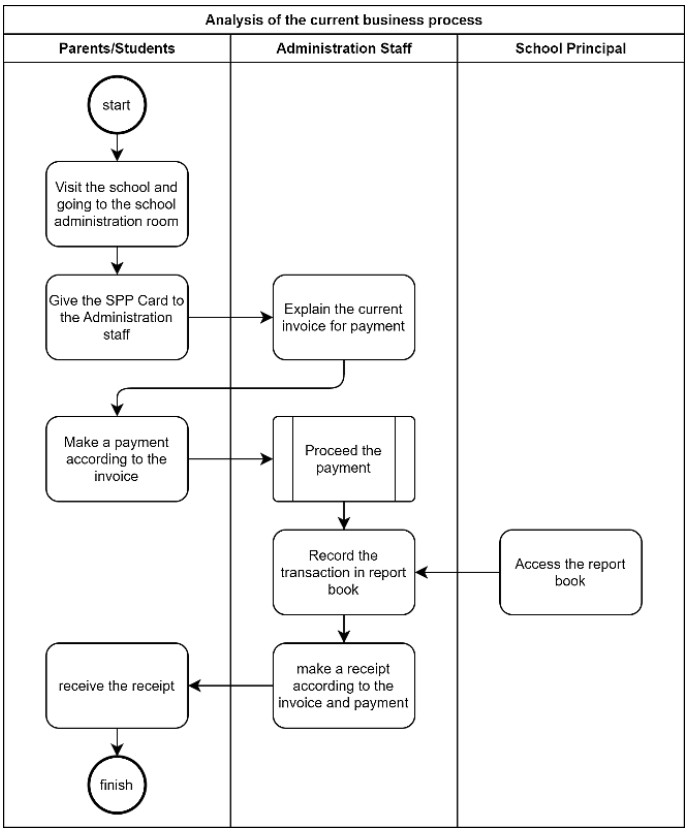

Figure 2. Requirement gathering and analysis

The recapitulation results of tuition fee payments collected so far show many unpredictable payment delays, making it difficult to carry out follow-up collections or other strategies so that tuition fee payments can be made on time. Use case and activity diagram were created based on the analysis results in Figure 2 to describe the previous analysis in a form that is easier to understand in designing web-based applications. Making this use case and activity diagram is the **Stage 2-Quick Design** in implementing the prototyping model, User Design. The following is a use case diagram for a web-based application:

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

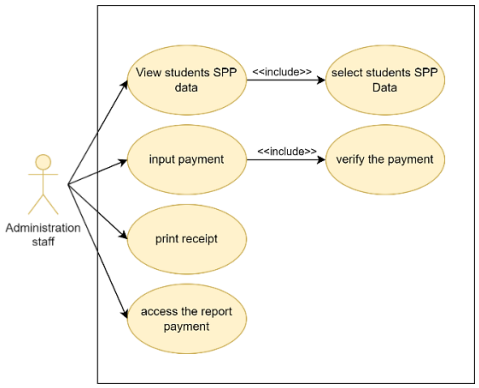

Figure 3. Use case diagram of web-based application

The use case diagram in Figure 3 shows the primary function of a web-based application user, namely Administration staff. The user has four use cases, namely view students' tuition fee data, input payment, print receipt, and access the report payment. Next, the use case diagram results in Figure 3 will be derived to the activity diagram in Figure 4. In the activity diagram section below, only one activity diagram, namely the tuition fee payment input process, will be shown as a examples of an activity diagrams. The activity diagram shown in Figure 4 describes the sequence of process activities in a web-based application. The activity diagram helps understand the sequence of activities and overall tuition fee payment business processes.

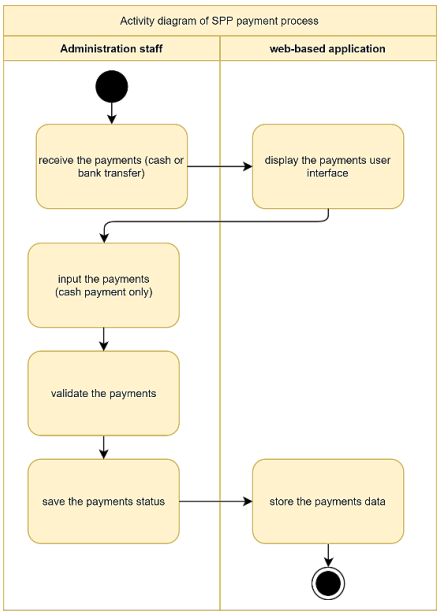

Figure 4. Activity diagram of tuition fee payment process

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

Then, from the activity diagram example in Figure 4, a prototype design is created as a web-based application mock-up, which is included in the **Stage 3-Build a Prototype**, of implementing the prototyping model. The results of the web-based application mock-up were evaluated according to user needs (stage 4-Initial User Evaluation) and immediately continued with stage 5, namely Refining Prototype. The following are some of the results of designing a web-based application user interface that has been refined:

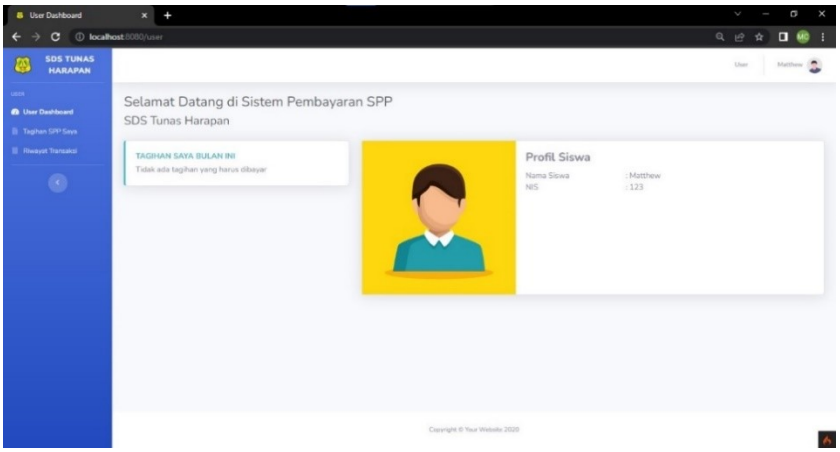

Figure 5. Tuition fee web-based application user interface

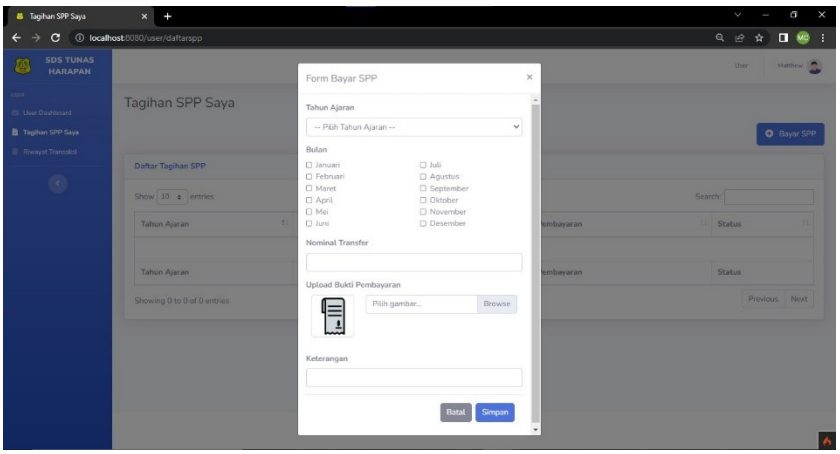

Figure 6. Select and input tuition fee payments

Figure 5 shows the dashboard page user interface of the tuition fee management web-based application. Some general information on the dashboard page includes User Dashboard, Tuition Billing, and Transaction History. Apart from that, users (administration staff) can also access the user profile menu, which is located in the

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

top right corner of the page with the username and icon. The user profile menu can be used to log out and manage user account information, such as changing passwords and other general information related to the user.

The main features of the administration staff actor can be seen in Figure 6. The select and input tuition fee payments feature has a drop-down button to select the academic year, a checklist radio button menu to choose the month, input tuition fee payments in the form of input (Integer data type), upload evidence transfer, and a description to provide information or news. After selecting the year, month, nominal input, and proof of transfer, you can save or cancel it.

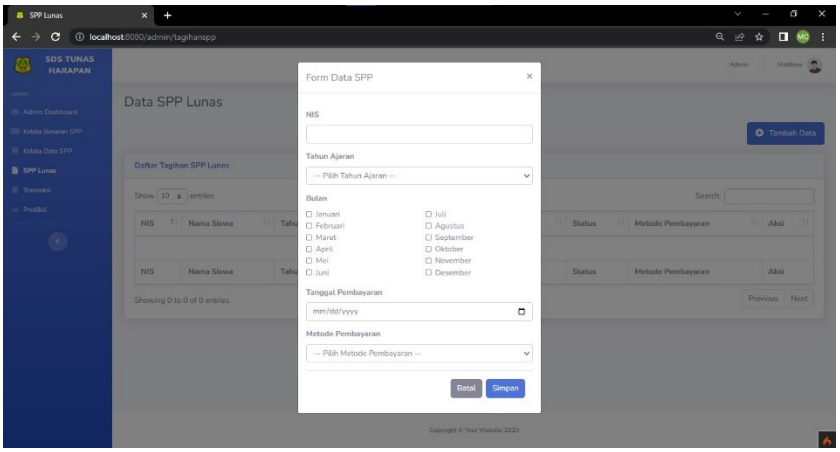

Figure 7. Access the report payments data

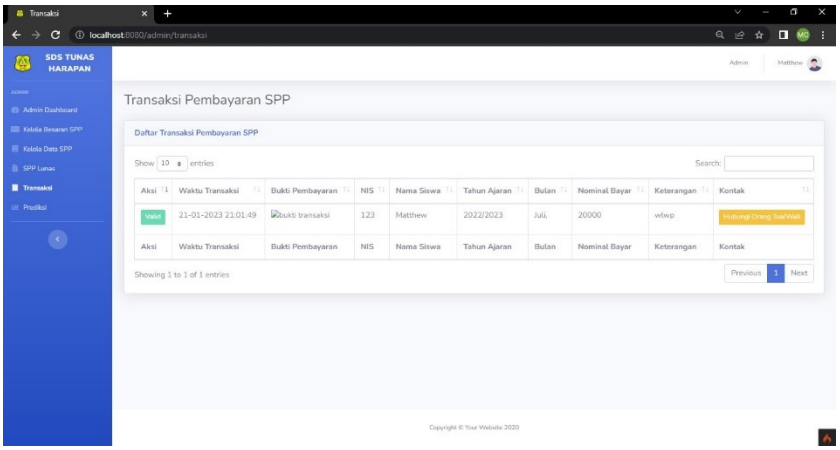

Figure 8. Transaction report of tuition fee payments

In the access, the reports payments data section shown in figure 8, user administration staff data sees the tuition fee data that has been paid. To view the

tuition fee data, you must input your Student Identification Number (NIS), academic year, month, payment date, and payment method (cash or transfer). If you have selected the criteria, the report results can be seen as shown in Figure 9. Figure 9 shows a report on the list of tuition payment transactions, which the School Principal can also access. In the transaction list report, there are several columns containing information: Payment status (valid/invalid), payment time, proof of payment, NIS, student name, academic year, month, payment amount, information, and contact. Then, apart from seeing the transaction list report, user administration staff can also see the history of tuition fee payment transactions, as shown in Figure 10.

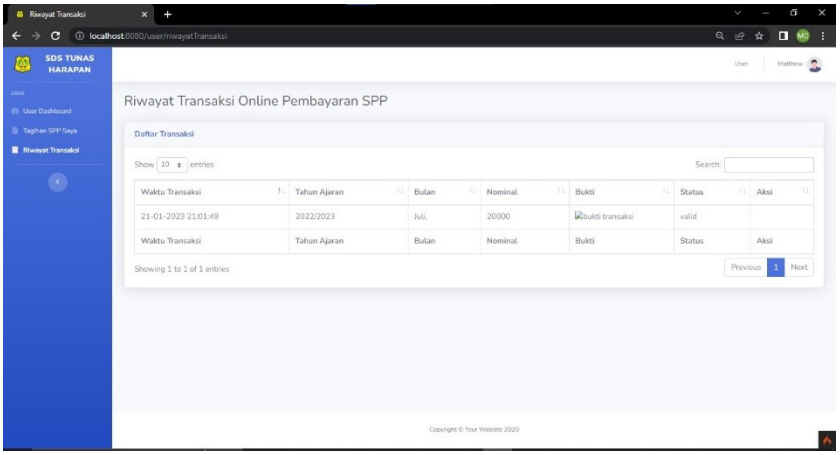

Figure 9. Transaction history of tuition fee payments

Apart from designing the web-based application, at the **Stage 4-Initial User Evaluation** and **Stage 5-Refining Prototype**, the Naïve Bayes (NB) algorithm was also implemented to predict the next tuition fee payment. The NB Algorithm Implementation Process begins with Data Preparation by collecting relevant data for predicting the next tuition fee payment. The data relates to students, parents, and tuition payment data in several previous periods. Furthermore, the data used in the NB algorithm simulation will be displayed in Table 1 and Table 2 as primary and secondary data, which can be seen in the tables below:

Table 1. Sample of tuition fee Payments data (Primary data)

No	Name	Gender	Date of Birth	Address	Previous payment status	Predicted payment status
1	Adit	M	02-Jan-2010	Kab. Tangerang	On-time	On-time
2	Aldiansyah Reza	M	04-Mar-2010	Kab. Tangerang	Overdue	Overdue
3	Intan Ayu Kencana	F	14-Apr-2010	Kab. Tangerang	On-time	On-time
4	Dika Dwi Febrian	M	23-Des-2010	Kab. Tangerang	On-time	On-time
5	Dita Silvi	F	30-Jan-2010	Kab. Tangerang	On-time	Overdue
6	Eri Setiawan	M	15-Aug-2010	Kab. Tangerang	Overdue	Overdue
7	Erika Yunadi	F	04-May-2010	Kab. Tangerang	Overdue	Overdue

No	Name	Gender	Date of Birth	Address	Previous payment status	Predicted payment status
8	Revina Ayutyas	F	03-Jun-2010	Kab. Tangerang	On-time	On-time
9	Kalvin	M	09-Sep-2010	Kab. Tangerang	On-time	On-time
10	Zahra Safitri	F	10-Nov-2010	Kab. Tangerang	On-time	On-time

Table 2. Sample of tuition fee Payments data (Secondary data)

No	Paretn age	Paretns occupation	Parents Education level	Dependant	Monthly income
1	35-44	Entrepreneur	High School	<2	3jt–5jt
2	35-44	Labourer	High School	2-4	1jt-3jt
3	<34	Civil Servants	Bachelor	<2	5jt
4	<34	Entrepreneur	Bachelor	<2	3jt-5jt
5	35-44	Entrepreneur	Bachelor	<2	3jt-5jt
6	<34	Labourer	High School	<2	1jt-3jt
7	35-44	Labourer	High School	<2	1jt-3jr
8	<34	Entrepreneur	Bachelor	2-4	3jt-5jt
9	35-44	Entrepreneur	Bachelor	<2	3jt-5jt
10	34-44	Entrepreneur	Bachelor	<2	3jt-5jt

After the data preparation stage, data selection is then carried out to select the data collected for use as a data mining process, separated from operational data or datasets. Apart from that, data cleaning is carried out to remove irrelevant data. Then, in the final stage, Data Processing is carried out to simulate predictions of tuition fee payments using the NB Algorithm, as seen in Table 3.

Table 3. Data Processing

No	Parents age	Dependant	Monthly income	Previous payment status	Predicted payment status
1.	35-44	<2	3jt-5jt	On-time	On-time
2.	35-44	2-4	1jt-3jt	Overdue	Overdue
3.	<34	<2	5jt	On-time	On-time
4.	<34	<2	3jt-5jt	On-time	On-time
5.	35-44	<2	3jt-5jt	On-time	Overdue
6.	<34	<2	1jt-3jt	Overdue	Overdue
7.	35-44	<2	1jt-3jt	Overdue	Overdue
8.	<34	2-4	3jt-5jt	On-time	On-time
9.	35-44	<2	3jt-5jt	On-time	On-time
10.	34-44	<2	3jt-5jt	On-time	On-time

The following is an example of calculating predictions for late payment of tuition fee using the NB Algorithm:

a. Parents age (PA) : <34
b. Dependant (D) : <2
c. Monthly Income (MI) : 3jt-5jt
d. Previous Payment Status (PPS) : On-time

X = (PA= "<34", D= "<2", MI= "3jt-5jt", PPS = "On-time")

Prior Probability

P(PPS= "On-time") = 6/10

P(PPS= "Overdue") = 4/10

Likelihood for Predicted payment status: On-time

P(Parents age="<34" | On-time) = 3/6
P(Parents Education level= "Bachelor" | On-time) = 5/6
P(Dependant= "<2" | On-time) = 5/6
P(Monthly Income= "3jt-5jt" | On-time) = 5/6
P(Previous Payment Status = "On-time" | On-time) = 6/6

Likelihood for Predicted Payment Status: Overdue

P(Parents age= "<34" | Overdue) = 1/4
P(Parents Education level= "Bachelor" | Overdue) = 1/4
P(Dependant= "<2" | Overdue) = 3/4
P(Monthly Income= "3jt-5jt" | Overdue) = 1/4
P(Previous Payment Status= "On-time" | Overdue) = 1/4

Posterior Probability

P(Predicted Payment Status= "On-time" | X) = 6/10×3/6×5/6×5/6×5/6×6/6= 0,174
P(Predicted Payment Status= "Overdue" | X) = 4/10×1/4×1/4×3/4×1/4×1/4= 0,001

Since the posterior probability for the next payment state to be on time is more significant than late, the following payment state is predicted to be on time. Prediction of Delay in Tuition Fee Payments is still in the development stage. It has yet to be fully implemented in this web-based tuition fee management application, but administration staff can access it in the available menu.

After the NB Algorithm simulation is carried out, the **Stage 6-Implement Product**, of this prototyping model will test the functionality of the tuition fee management web-based application using the UAT method, black box testing model. Table 4 below shows the results of testing features on the web-based application that can be used by administration staff according to the use case diagram and activity diagrams:

Table 4. Tuition Fee Management User Acceptance Test results

Features	Test Case	Feedback User	Status
View Students tuition fee Data	View the student's data by inputting the NIS	The student's data has been displayed according to the NIS	valid
Input payment	Input the payment that has been received and verify the payment.	Users can input and verify the payment.	valid
Print receipt	Print the receipt after verifying the payment	Web-based application can provide the receipt (print out).	valid
Access the payment reports	View the transaction history	Users can access the payment reports by input a spesific criteria.	valid

The testing results using the black-box testing model show that the resulting web-based application is following the needs of users, namely administration staff. The suitability of the test results to user needs can be seen in the test scenario and user feedback. The four test case results have been validated by administrative staff.

4. CONCLUSION

This research produces a web-based education fee management application that can optimize the process of recording education fee payments, view the history of payment transactions and feature predictions of late payment of education fees. The payment proof upload feature can make it easier for administrative officers to store tuition payment transactions and produce payment reports, which are stored in a database and can be accessed by the School Principal.

Implementing the Naive Bayes algorithm in predicting potential payment delays can also be done with the output status "On-time" and "Overdue". However, the accuracy of these predictions still needs to be improved in future research. The conclusions obtained in this research are also strengthened by the success of functionality testing users through black box testing. The four features tested by users are valid, so this website-based application is suitable and ready for tuition fee management.

ACKNOWLEDGEMENTS

We extend our heartfelt gratitude to Universitas Multimedia Nusantara for their invaluable support, pivotal in completing this research endeavor. Their substantial contribution was instrumental in achieving our objectives, and we are deeply grateful for their unwavering assistance.

REFERENCES

[1] E. Milkova, S. Pekarkova, and A. B. M. Salem, "Information and Communication Technology in Education-Current Trends," *MATEC Web Conf.*, vol. 76, no. January, 2016, doi: 10.1051/matecconf/20167604022.

[2] C. D. Elviera, D. Irawan, and D. N. Syafrina, "Pengembangan Tata Usaha Sekolah Berbasis Teknologi Informasi," *MUKADIMAH J. Pendidikan, Sejarah, dan Ilmu-ilmu Sos.*, vol. 3, no. 1, pp. 1–6, 2019, doi: 10.30743/mkd.v3i1.676.

[3] A. A. Putri, "Administrasi Keuangan," *Univ. Negeri Padang*, p. 2, 2019.

[4] Y. Zhu, "A data driven educational decision support system," *Int. J. Emerg. Technol. Learn.*, vol. 13, no. 11, pp. 4–16, 2018, doi: 10.3991/ijet.v13i11.9582.

[5] D. Y. Kim and S.-A. Kim, "An exploratory model on the usability of a prototyping-process for designing of Smart Building Envelopes," *Autom.*

Constr., vol. 81, 2017, doi: 10.1016/j.autcon.2017.03.012.

[6] E. W. Fridayanthie, H. Haryanto, and T. Tsabitah, "Penerapan Metode Prototyping model Pada Perancangan Sistem Informasi Penggajian Karyawan (Persis Gawan) Berbasis Web," *Paradig. - J. Komput. dan Inform.*, vol. 23, no. 2, pp. 151–157, 2021, doi: 10.31294/p.v23i2.10998.

[7] A. Kurniawan, M. Chabibi, and R. S. Dewi, "Pengembangan Sistem Informasi Pelayanan Desa Berbasis Web Dengan Metode Prototyping Pada Desa Leran," *JURIKOM (Jurnal Ris. Komputer)*, vol. 7, no. 1, p. 114, 2020, doi: 10.30865/jurikom.v7i1.1863.

[8] T. Rosandy, "Perbandingan Metode Naive Bayes Classifier Dengan Metode Decision Tree (C4.5) Untuk Menganalisa Kelancaran Pembiayaan (Study Kasus : Kspps / Bmt Al-fadhila," *J. Teknol. Inf. Magister Darmajaya*, vol. 2, no. 01, pp. 52–62, 2016.

[9] F. A. Lestari, L. Efrizoni, E. Ali, and R. Rahmiati, "Sistem Klasifikasi Pengaduan Masyarakat Pada BPJS Ketenagakerjaan Menggunakan Algoritma Naïve Bayes Berbasis Mobile," *Build. Informatics, Technol. Sci.*, vol. 4, no. 1, pp. 217–226, 2022, doi: 10.47065/bits.v4i1.1685.

[10] D. Purnomo, "Model Prototyping Pada Pengembangan Sistem Informasi," vol. 2, no. 2, pp. 54–61, 2017.

[11] P. Ogedebe and B. P. Jacob, "Prototyping : A Strategy to Use When User Lacks Data Processing Experience," 2012. [Online]. Available: https://api.semanticscholar.org/CorpusID:6851479

[12] R. McLeod and G. P. Schell, *Management information systems*. Upper Saddle River, N.J. SE - xxix, 447 pages : illustrations ; 26 cm: Pearson/Prentice Hall Upper Saddle River, N.J., 2007. doi: LK - https://worldcat.org/title/64453371.

[13] C. Prof. Dr. Sri Mulyani, Ak., "Metode Analisis dan Perancangan Sistem," 2016. https://books.google.co.id/books?id=SbrPDgAAQBAJ&printsec=copyright&hl=id#v=onepage&q&f=false (accessed Sep. 22, 2023).

[14] A. Nugroho, "Rekayasa Perangkat Lunak Menggunakan UML dan Java," 2009. https://books.google.co.id/books?id=uly0ekGR_X4C&printsec=copyright&hl=id#v=onepage&q&f=false (accessed Sep. 22, 2023).

[15] C. C. Aggarwal and C. X. Zhai, *Mining text data*, vol. 9781461432234. 2013. doi: 10.1007/978-1-4614-3223-4.

[16] Zaidir, "Pengujian Sistem Informasi Pengelolaan Kegiatan Satuan Tugas Penanganan Masalah Perempuan Dan Anak Dengan Metode Black-Box Test Dan User Acceptance Test," *Semin. Nas. UNRIYO*, pp. 281–288, 2020.

[17] Sambas and I. Ripai, "Implementasi dan User Acceptance Test (UAT) Aplikasi Integrated Library System (Inlis Lite) Di MTS Negeri 7 Kuningan," *ICT Learn.*, 2022, [Online]. Available:

https://api.semanticscholar.org/CorpusID:259247776
[18] I. Afrianto, A. Heryandi, A. Finadhita, and S. Atin, "User Acceptance TestForDigital Signature Application In Academic Domain To Support The Covid-19 Work From Home Program," *Int. J. Inf. Syst. Technol.*, vol. 5, no. 3, pp. 270–280, 2021, doi: 10.30645/ijistech.v5i3.