Software To Predict Maternal and Child Health Risks with Machine Learning

Fitriana Fitriana[1], Zulkifli Zulkifli[2,*], Sri Rahayu[3]

[1]*Department of Midwifery, Faculty of Health, Aisyah University, Pringsewu 35372, Indonesia*

[2]*Department of Informatics Engineering, Faculty of Tehcnology and Informatics, Aisyah University, Pringsewu 35372, Indonesia*

[3]*Department Master of Midwifery, Faculty of Health, Aisyah University, Pringsewu 35372, Indonesia*

(Received: July 20, 2025; Revised: September 15, 2025; Accepted: December 15, 2025; Available online: January 14, 2026)

Abstract

Maternal healthcare is essential to safeguard the well-being of mothers and their children during pregnancy, childbirth, and the postpartum period. This study aims to develop and evaluate a software application that predicts maternal and child health risks using the Naive Bayes algorithm as the primary predictive model. The research focuses on enhancing early detection accuracy while ensuring that the system remains practical and affordable for healthcare use. An Android-based software application was designed to analyze seven key maternal health variables patient code, age, systolic and diastolic blood pressure, blood glucose level, body temperature, and heart rate to generate a single output representing the predicted risk level. To validate the effectiveness of the Naive Bayes model, its performance was benchmarked against Neural Network (NN) and Random Forest (RF) algorithms using a dataset of 1,015 maternal health records obtained from Kaggle. Model performance was assessed based on accuracy, precision, recall, and F1-score. The Naive Bayes model achieved an accuracy of 63%, performing comparably to Random Forest (67%) and better than Neural Network (48%). The estimated software development cost was IDR 1,635,913, confirming the feasibility of producing a cost-effective health application for risk prediction. The proposed Naive Bayes–based software offers an accurate, low-cost, and accessible solution for early detection of maternal and child health risks. It provides a valuable decision-support tool for healthcare providers and pregnant women, enabling timely intervention and improved maternal care outcomes. The findings demonstrate the potential of machine learning–based software in improving maternal health management. Future work may focus on enhancing prediction accuracy through larger datasets and integrating additional clinical and demographic variables.

Keywords: Accuracy Level, Child Risk, Machine Learning, Maternal, Prediction

1. Introduction

As of 2020, approximately 800 women die each day from pregnancy and childbirth-related causes that are largely preventable. Sustainable Development Goal (SDG) 3.1 aims to reduce maternal mortality to fewer than 70 deaths per 100,000 live births by 2030. About 94% of maternal deaths occur in low-resource settings and are mostly avoidable. Maternal health plays a critical role in perinatal outcomes, and many complications can be prevented during the first trimester [1]. For example, rising obesity rates among women are connected to increased risks of fetal macrosomia, gestational diabetes [2], hypertensive disorders during pregnancy, and the birth of large-for-gestational-age (LGA) infants [3]. Furthermore, declining birth rates in developed countries have led to higher maternal age, which is associated with greater risks of adverse pregnancy outcomes [4].

Assessing maternal health risks is crucial for protecting the well-being of pregnant women and reducing both maternal mortality as well as morbidity. Maternal health comprises the mental, emotional, and social well-being of women during pregnancy, childbirth, as well as the postpartum period [5]. It includes medical conditions, lifestyle factors, access to healthcare as well as various social and economic influences [6].

Protecting the health of both mother and child requires continuous attention to maternal care from pregnancy through the postpartum period [7]. At every stage, various health risks can seriously impact maternal and fetal outcomes [8].

***Corresponding author: Zulkifli Zulkifli (zulkifli@aisyahuniversity.ac.id)**

DOI: https://doi.org/10.47738/jads.v7i1.1088

Obstetric emergencies such as hemorrhage, hypertensive disorders, sepsis, abortion, obstructed labor, ectopic pregnancy, and embolism pose significant as well as immediate threats to maternal health [9]. Moreover, postpartum hemorrhage remains the leading cause of maternal death worldwide, with approximately 24% of hemorrhage-related deaths occurring during pregnancy [10]. Improving emergency obstetric care during labor is a major strategy for prevention [11]. Healthcare delivery systems should be strengthened to effectively reduce maternal mortality. Despite ongoing global efforts, maternal mortality remains a critical concern, particularly in regions with limited access to healthcare services [12]. Improving maternal and child healthcare services requires the development of a more accurate and cost-effective early screening method [13]. A proposed solution includes designing software that uses the Naive Bayes algorithm to predict potential health risks for both mothers and children [14].

Numerous studies have explored maternal health risks, including a 2025 study by Pavagada [12], which showed the importance of closely monitoring maternal health. This study examined the application of Machine Learning (ML) models for predicting maternal health risks, with the Ensemble Bagged Trees method achieving the highest accuracy at 84.12%, outperforming other models. The findings showed the potential of ML to provide timely and accurate risk assessments for expectant mothers. In developing software to predict maternal and child health risks, the literature review will incorporate studies on multi-algorithm frameworks [15], data mining, and ML applications [16]. For example, a 2024 study by Makkiyah [17] explored the development of an application using a multi-algorithm method to predict diabetes status. Another relevant study by Nia [18] evaluated the use of Artificial Intelligence (AI) in disease diagnosis and prediction. The finding showed that AI could reduce physician workload, minimize diagnostic errors and time, as well as improve the total effectiveness of disease detection.

2. Literature Review

This study will adopt an experimental method to develop software that predicts maternal and child health risks using the Naive Bayes algorithm. The accuracy of the model will be evaluated by comparing it with other algorithms, including RF and NN. The aim is to support risk prediction during pregnancy and facilitate early identification of potential complications. Several previous studies have addressed maternal and child health risk prediction. For instance, a 2024 study by Jamel [19] focused on maternal healthcare services throughout pregnancy, childbirth, and the postpartum period. The study showed the importance of early risk detection and maternal vulnerability during these stages. It proposed a prediction model that began with Principal Component Analysis (PCA) to extract major features, followed by a stacked ensemble voting classifier combining ML and a deep learning model. The PCA-based model achieved strong performance—98.25% accuracy, 99.17% precision, 99.16% recall, and a 99.16% F1-score, outperforming several state-of-the-art methods. Another significant study by [4] introduced the Mud Ring Algorithm (MRA) for parameter optimization in maternal health risk prediction. In the first phase, MRA optimized a Support Vector Machine (SVM) model, tested across 13 real-world datasets. In the second phase, the study addressed class imbalance using a crossover oversampling strategy. Additional models such as RF and K-Nearest Neighbor were also tested. The MRA-enhanced models showed substantial performance gains, where accuracy increased by 11.8% for MRA-SVM, 9.11% for MRA-RF, and 17.08% for MRA-KNN. These models outperformed six other optimization methods across metrics such as Accuracy, G-mean, F-measure, MCC, and Kappa [4]. A 2023 study by [6] explored the use of Exploratory Data Analysis (EDA) in predicting maternal health risks. This study conducted a comprehensive analysis of relevant datasets to build a robust ML model and incorporate Explainable AI (XAI) methods to interpret the decision-making process of top-performing algorithms [6]. Following the discussion, a 2025 study by [20] introduced molecular biomarker profiling as a developing method in maternal-fetal health. By analyzing maternal blood for biomarkers, the method provided perceptions of placental function, enabling early diagnosis and intervention. The study signified that traditional tools such as ultrasound and Doppler imaging might not detect placental dysfunction before clinical symptoms appear. Since the placenta plays a critical role in fetal development, early detection is important for preventing serious complications such as preeclampsia [20].

Previous studies have explored various approaches to improving maternal health risk prediction. For instance, [19] applied PCA with a stacked ensemble classifier to enhance feature extraction and achieve high predictive accuracy. While this highlights the potential of complex ensemble and dimensionality-reduction methods, such techniques require substantial computational resources and are less suitable for real-time healthcare applications in low-resource settings. In contrast, the present study adopts the Naive Bayes algorithm, prioritizing simplicity, interpretability, and low computational cost for deployment on mobile devices. Similarly, [4] introduced the MRA to optimize machine learning parameters across multiple datasets. Although their method significantly improved classification performance, it involves advanced optimization processes not easily integrated into lightweight applications. This study, therefore, emphasizes practicality and ease of implementation over complex optimization, aiming to produce an accessible tool for healthcare providers.

Most existing studies on maternal and child health risk prediction have focused on algorithm development, with limited progress toward practical software applications. To address this gap, the proposed software aims to function as an early diagnostic tool for hospitals, healthcare providers, and pregnant women in assessing potential health risks during pregnancy. The application will be developed for the Android platform and use seven input variables, namely patient code, age, systolic blood pressure, diastolic blood pressure, blood glucose level, body temperature, and heart rate, to generate a single output known as the predicted risk level.

3. Methodology

The following were the stages of software to predict maternal and child health risks with Naive Bayes, NN, and RF. Figure 1 showed the stages of software development predict maternal and child health risks using Naive Bayes, NN, and RF. The initial steps included problem identification, problem analysis, design, and coding, followed by application implementation [21]. The process comprised two main inputs, namely data from pregnant women and medical records based on seven variables. The outcomes included a dataset on maternal and child health risks as well as a comparison of prediction results for maternal and child health status using NN, RF, and NN algorithms.

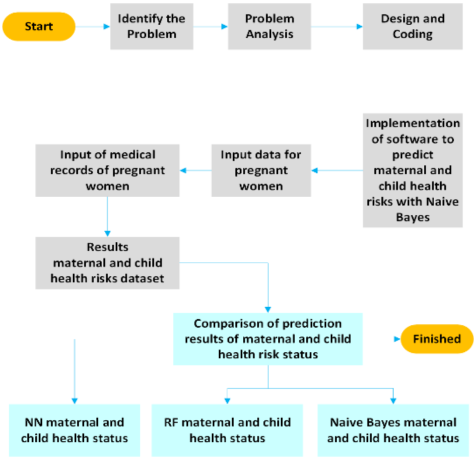
Start
Identify the Problem
Problem Analysis
Design and Coding
Implementation of software to predict maternal and child health risks with Naive Bayes
Input data for pregnant women
Input of medical records of pregnant women
Results maternal and child health risks dataset
Comparison of prediction results of maternal and child health risk status
Finished
NN maternal and child health status
RF maternal and child health status
Naive Bayes maternal and child health status

Figure 1. Research flow design

Figure 2 showed the study framework of software designed to predict maternal and child health risks using the Naive Bayes method. This framework was flexible and could be developed according to different needs. The stages included pregnant women or doctors logging in and when the login was successful, experts could input data as well as medical records of pregnant women with variables such as patient code, age, systolic/diastolic blood pressure, blood glucose, body temperature, and heart rate. Based on these seven input variables, the system produced a risk level status using Naive Bayes. This output formed the maternal and child health risk results, which were stored in a dataset database. The dataset was then exported as a CSV file and loaded into Google Colab, where scripting was performed using Python. To measure accuracy, the expert used the orange widget tool. From the above process, prediction results were obtained from NN [21], RF [22], and Naive Bayes [23]. While the current implementation relies on Google Colab for model training and evaluation, future work may integrate these processes directly into the Android system or a secure web server. This enhancement would allow real-time risk prediction and reduce dependency on manual CSV data exchange, improving system efficiency and scalability. Although Orange was primarily chosen for its user-friendly interface and compatibility with scikit-learn, future research may incorporate additional platforms such as Weka or TensorFlow to enable deeper algorithmic customization and scalability testing. Cross-validation of results using multiple tools would further strengthen methodological transparency and reproducibility.

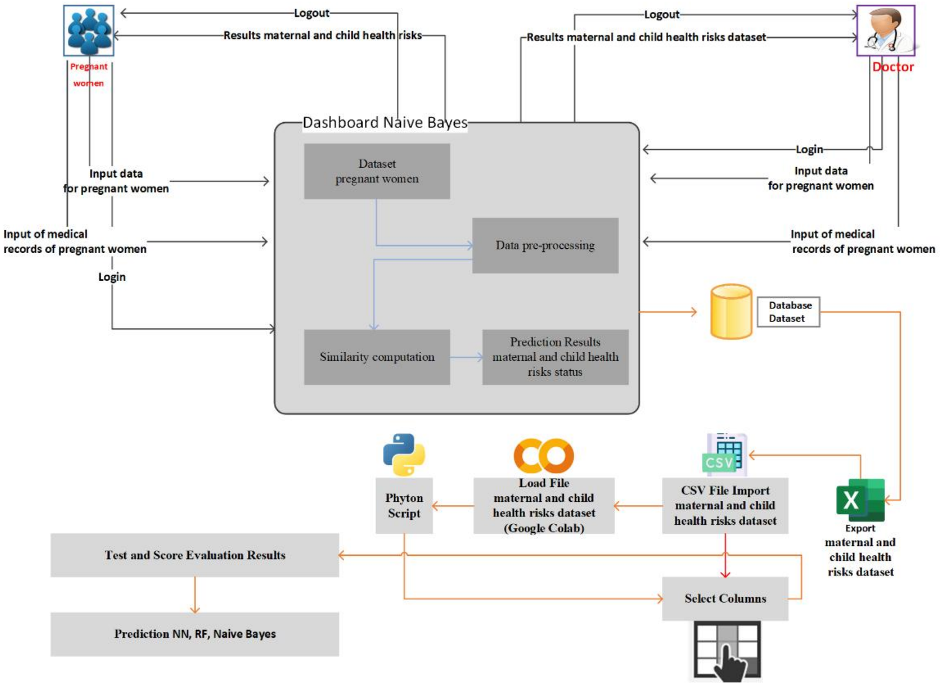
Logout
Results maternal and child health risks
Pregnant women
Logout
Results maternal and child health risks dataset
Doctor
Dashboard Naive Bayes
Dataset pregnant women
Data pre-processing
Similarity computation
Prediction Results maternal and child health risks status
Login
Input data for pregnant women
Input of medical records of pregnant women
Input data for pregnant women
Input of medical records of pregnant women
Login
Database Dataset
CSV
Phyton Script
Load File maternal and child health risks dataset (Google Colab)
CSV File Import maternal and child health risks dataset
Export maternal and child health risks dataset
Test and Score Evaluation Results
Select Columns
Prediction NN, RF, Naive Bayes

Figure 2. Study framework software to predict maternal and child health risks with Naive Bayes, NN, and RF

4. Results and Discussion

The stages of the software developed to predict maternal and child health risks using Naive Bayes began with generating a dataset by inputting data and medical records of pregnant women based on seven variables. The system then predicted the risk level status using NN, RF, and Naive Bayes algorithms. At this stage, the problem was identified as the need to develop software to predict maternal and child health risks using the Naive Bayes method, based on seven input variables and one output variable.

4.1. Problem Identification

Problem identification was conducted during this stage, the development of software capable of predicting maternal and child health risks using the Naive Bayes method is essential to support effective decision-making and the implementation of appropriate policies in healthcare services for pregnant women. The resulting predictive model should be designed to ensure its applicability within real healthcare settings, enabling practitioners to utilize it as part of routine maternal and child health services. Furthermore, this study includes an evaluation of prediction accuracy by comparing the performance of the Naive Bayes algorithm with NN and RF, thereby providing comprehensive insights into the model's reliability and suitability for clinical use [24], [25]. The problem analysis in this study is communicating with tool users to understand the software expectations from both pregnant women and doctors.

4.2. Design Software to Predict Maternal and Child Health Risk with NB, NN, and RF

The software used during this analysis was developed for the Android platform [26]. Several interfaces shown in figure 3 were presented in the following image.

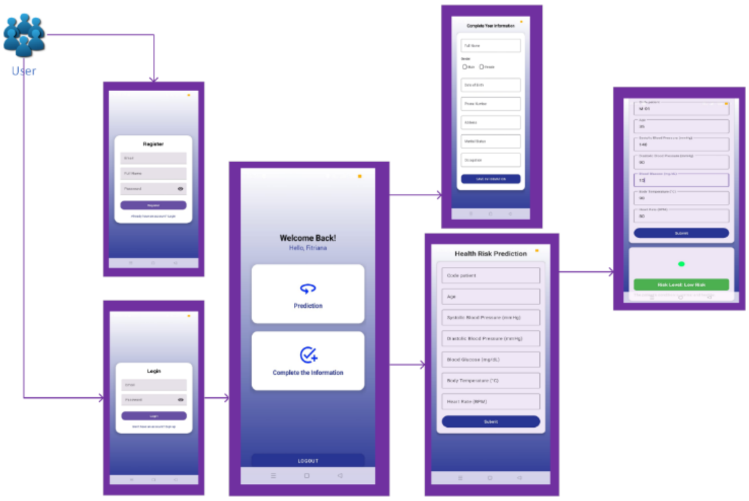

Figure 3. Software display to predict maternal and child health risks with Naive Bayes, NN, and RF

4.3. Practical Complexity of Software to Predict Maternal and Child Health Risks

At this stage, effort estimation was conducted using Use Case Point (UCP) method, which served as the basis for determining the required time, personnel, and costs. The process began with calculating the Unadjusted Actor Weight (UAW) [27], which was obtained by categorizing the actors of the system and assigning a weight to each category based on each role in the system [28]. After determining the weights for actors and use cases, the next step was to calculate UAW and the Unadjusted Use Case Weight (UUCW). In this study, actors refer to the primary entities that interact with the maternal health risk prediction software, including pregnant women as end users, doctors or midwives as data input and review agents, and the system administrator responsible for managing user access and database updates.

The results of UAW and UUCW calculation were presented in table 1. Average (Weight = 2) represents actors that interact through a standard user interface with limited validation. These classifications determine the UAW value by multiplying the number of actors in each category by the corresponding weight. A similar process was applied to the UUCW calculation, where each use case (such as “Input Patient Data,” and “View Risk Prediction,”) was assigned a complexity level and weighted accordingly. The next stage was to calculate the value of the unadjusted use case point (UUCP) which was the result of adding the total UAW and UUCW values. The resulting subtotals are then summed, yielding a total UAW value of 6. In parallel, the UUCW calculation assigns weights to the use cases associated with each actor and multiplies them by their respective quantities. The sum of all UUCW subtotals results in a total value of 40. Finally, the UUCP value is obtained by adding the total UAW and UUCW, producing a final UUCP score of 46, which represents the unadjusted functional size of the system prior to the application of technical and environmental adjustment factors.

Table 1. Calculating UAW value

Actor	Type	Weight (UAW)	Qty	Subtotal (UAW)	Weight (UUCW)	Qty	Subtotal (UUCW)
System Administrator	Simple	2	1	2	10	1	10
Pregnant Woman	Average	2	1	2	10	1	10
Doctor/Midwife	Complex	2	1	2	15	1	20
Total				**6**			**40**

Table 2 presents the calculation of the Technical Complexity Factor (TCF) used to adjust the Unadjusted Use Case Points based on the system’s technical characteristics. The table lists 13 technical factors (T1–T13), such as distributed system requirements, response time, security features, and ease of use. Each factor is assigned a predefined weight that reflects its relative importance and a score ranging from 0 to 5 that represents the degree to which the factor applies to the system. The product of the weight and the score (B * S) is calculated for each factor, and all results are summed to obtain the total technical factor (TF) value of 53.5. The TCF is then calculated using the standard formula TCF = 0.6 + (0.01 * TF). Substituting the obtained TF value results in a TCF of 1.135. This TCF value serves as a technical adjustment multiplier that reflects the overall technical complexity of the system and is later applied to refine the system size estimation.

Table 2. Calculating TCF value

No	Technical factor	Weight	Score (0–5)	B * S
T1	Distributed system required	2	4	8
T2	Response time	2	3	6
T3	End-user efficiency	1	4	4
T4	Complex internal processing required	1	2	2
T5	Reusable code	1	3	3
T6	Easy to Install	0.5	3	1.5
T7	Easy to use	0.5	4	2
T8	Portable	2	3	6
T9	Easy to Change	1	5	5
T10	Concurrent	1	4	4
T11	Security features	1	4	4
T12	Access for Third Parties	1	4	4
T13	Special training required	1	4	4
Total TF				53.5
Total TCF (0.6 + (0.01 * TF))				1.135

Table 3 shown the calculation of the Environmental Complexity Factor (ECF), which reflects the influence of environmental and team-related conditions on the software development process. It consists of eight environmental factors (E1–E8), including familiarity with the project, application and object-oriented programming experience, analyst capability, team motivation, requirement stability, staff availability, and programming language difficulty. Each factor is assigned a specific weight and a score ranging from 0 to 5 based on its relevance to the project. The product of the weight and the score (B * S) is calculated for each factor, and all values are summed to obtain a total Environmental Factor (EF) score of 31. The ECF is then calculated using the formula ECF = 1.4 + (−0.03 * EF). Substituting the EF value into the formula yields an ECF of 0.47. This ECF value represents the overall impact of the development environment on the project and is used as an adjustment multiplier to refine the final use case point estimation.

Table 3. Calculating ECF value

No	Environment Factor	Weight	Score (0-5)	B * S
E1	Familiarity with the project	1.5	4	6
E2	Application experience	0.5	4	2
E3	Oo programming experience	1	4	4
E4	Lead analyst capability	0.5	4	2
E5	Motivation	1	4	4
E6	Stable requirements	2	3	6
E7	Part time staff	1	3	3
E8	Difficult programming language	1	4	4
Total EF				31
Total ECF (1.4 + (-0.03 * EF))				0.47

After obtaining the TCF and ECF values, the next step is to calculate the Use Case Points (UCP) to estimate system size and development effort. The UCP is computed by multiplying the Unadjusted Use Case Points (UUCP) by the TCF and ECF values. With a UUCP value of 46, the calculation yields UCP = 46 * 1.135 * 0.47 = 24.54. Finally, the development effort is estimated by multiplying the UCP value by a productivity factor of 20 hours per UCP, resulting in an estimated effort of approximately 490.77 person-hours, which represents the total development effort required for the system.

While the UCP methodology offered a systematic approach for estimating development effort, it does not fully account for contextual factors that may influence estimation accuracy. Elements such as project risk, requirement volatility, and variations in team experience or technical expertise can significantly affect productivity levels and cost outcomes. In this study, the estimation relied on standard Technical and Environmental Complexity Factors, which partially reflect environmental stability and team capability. However, a more comprehensive risk assessment incorporating factors such as project uncertainty, developer familiarity with machine learning frameworks, and software validation complexity could enhance the reliability of future estimations. Integrating risk sensitivity or hybrid estimation models, such as UCP combined with COCOMO II or Function Point Analysis, is recommended to improve precision in software project planning, particularly for healthcare applications where system reliability is critical. The next step was to calculate the estimated time and costs required for developing the software shown in table 4.

Table 4. Practical complexity of software to predict maternal and child health risks with NB, NN, and RF

Effort	**490.77**	**Productivity Factor**				
Productivity Factor (Karner. 1993). PF	28	20	10	5	4	2
Duration (hours)	13741.7	9815.8	4907.4	2453.7	1963.10	981.5
Duration (week)	286.28	204.49	102.24	51.12	40.90	20.45
Duration (month)	37.55	26.82	13.41	6.70	5.36	2.68

Table 5 shown the estimation of development time and cost based on different productivity factor (PF) scenarios, namely PF = 2 and PF = 4. The table identifies two project roles involved in the development process, namely a System Analyst and a Programmer, with one person assigned to each role. For the PF = 2 scenario, each role is assumed to work 2 hours per day, equivalent to 4 hours per week. Based on the previously calculated total effort, this workload results in a total of 82 working hours per role. With an hourly salary rate of 10,000, the estimated cost for each role is 817,957, producing a total project cost of 1,635,913. In the PF = 4 scenario, the assumed workload increases to 8 hours per week for each role, reflecting a higher productivity or more intensive work allocation. Under this condition, the total workload for each role becomes 327 hours. Using the same hourly salary rate of 10,000, the estimated cost for each role is 3,271,827, resulting in a total project cost of 6,543,653. Overall, the table demonstrates how variations in the productivity factor and weekly workload significantly affect the estimated project duration and development cost, providing alternative planning scenarios for resource allocation and budgeting.

Table 5. Calculate estimated time and cost

No	**Role**	**Qty**	**Workload (hours/day)**	**Workload (hours/week)**	**Total (hours/week)**	**Workload Total**	**Salary (Per Hour)**	**Sum Salary**
Calculate estimated time and cost with PF=2								
1	System Analyst	1	2	4	4	82	10,000	817,957
2	Programmer	1	2	4	4	82	10,000	817,957
	Total	2						1,635,913
Calculate estimated time and cost with PF=4								
1	System Analyst	1	2	8	8	327	10,000	3,271,827
2	Programmer	1	2	8	8	327	10,000	3,271,827
	Total	2						6,543,653

The selection of PF = 2 in the final estimation represents an optimistic but feasible scenario for a small, well-scoped academic project with low environmental uncertainty. In professional or industrial settings, higher PF values (e.g., 5–10) would likely be more appropriate to account for larger teams, evolving requirements, and higher validation demands. Future studies may refine PF calibration using empirical productivity data from comparable health-informatics projects to improve estimation accuracy.

4.4. Dataset Database

The dataset used to evaluate the accuracy of the maternal and child health risk prediction software using RF, Naive Bayes, and NN algorithms was shown in table 6 [29]. The dataset sourced from Kaggle contains 1,015 anonymized maternal health records. Prior to model training, data quality was assessed for completeness, outliers, and missing values. Normalization was applied to ensure consistent feature scales. Ethical considerations regarding data privacy and secondary use were also reviewed, as the dataset is publicly available and anonymized in accordance with Kaggle's open-data policy. The seven input variables age, systolic and diastolic blood pressure, blood glucose level, body temperature, and heart rate were selected based on a combination of clinical relevance and preliminary data analysis. Consultation with maternal health experts confirmed that these indicators are among the most critical physiological variables routinely measured during antenatal visits and directly linked to maternal morbidity risks such as preeclampsia and gestational diabetes. To confirm their relevance, pairwise correlation and feature importance analysis (using Random Forest) were conducted, showing that these six physiological variables had the highest information gain relative to the target "risk level" class. Other available attributes were excluded due to redundancy or low correlation. Patient code was retained as an identifier but omitted from the training process.

Table 6. Dataset maternal and child health risks

Code Patient	Age	Systolic Blood Pressure	Diastolic Blood Pressure	Blood Glucose	Body Temperature	Heart rate	Risk Level Status
M-0001	25	130	80	15	98	86	high risk
M-0002	35	140	90	13	98	70	high risk
M-0003	29	90	70	8	100	80	high risk
M-0004	30	140	85	7	98	70	high risk
M-0005	35	120	60	6.1	98	76	low risk
M-0006	23	140	80	7.01	98	70	high risk
M-0007	23	130	70	7.01	98	78	mid risk
M-0008	35	85	60	11	102	86	high risk
M-0009	32	120	90	6.9	98	70	mid risk
M-0010	42	130	80	18	98	70	high risk
M-0011	23	90	60	7.01	98	76	low risk
M-0012	19	120	80	7	98	70	mid risk
M-0013	25	110	89	7.01	98	77	low risk
M-0014	20	120	75	7.01	100	70	mid risk
M-0015	48	120	80	11	98	88	mid risk

The unequal distribution of risk classes in the dataset may have influenced model performance, particularly for minority categories such as high risk. While stratified sampling was used to maintain proportional representation, the absence of balancing methods such as Synthetic Minority Oversampling (SMOTE) or cost-sensitive weighting could have limited the model's ability to generalize to underrepresented cases. Future research should investigate the use of these techniques or collect a more balanced dataset to enhance sensitivity and ensure equitable prediction accuracy across all risk levels.

4.5. Application of Neural Network Algorithm

Table 7 showed a sample of the normalized dataset during the process. The normalization process converted the original test data to a range between 0.1 and 0.9, using a formula designed to accommodate the sigmoid activation function, which operated with values greater than 0 [30]. Prior to training the neural network, all continuous input features (age, systolic and diastolic blood pressure, blood glucose, body temperature, heart rate) were rescaled using min–max normalization to the range [0.1, 0.9]. For each feature we computed the training-set minimum x_{min} and maximum x_{max} and applied the transform:

$$x_{scaled} = 0.1 + 0.8 \times \frac{x - x_{min}}{x_{max} - x_{min}} \tag{1}$$

The factor 0.8 equals $0.9 - 0.1$. This range was chosen to keep inputs away from the extreme tails of the sigmoid activation function used in the network, reducing saturation and preserving gradient magnitude during training. All x_{min} and x_{max} values were calculated only on the training partition and then reused to transform the validation and test partitions to prevent data leakage. If a feature had zero variance in the training set (i.e. $x_{max} = x_{min}$), that feature's scaled value was set to 0.5 for all samples. After scaling we clipped values to $[0.9 - 0.1]$ to guard against numerical drift. Identifier fields (e.g., patient code) were excluded from the feature set and the target risk class was encoded as label-encoded for single-output classification. Alternative preprocessing options (standardization, robust scaling, or batch normalization) were considered but min–max scaling to [0.1,0.9] was adopted because of the sigmoid activations and the small number of features.

Table 7. Dataset normalization

Code Patient	Age	Systolic Blood Pressure	Diastolic Blood Pressure	Blood Glucose	Body Temperature	Heart rate	Risk Level Status
M-0001	0.5333	0.4862	0.1125	0	0.0656	high risk	0.2
M-0002	0.6222	0.6431	0.0875	0	0.0523	high risk	0.3333
M-0003	0.1777	0.3294	0.0250	0.32	0.0606	high risk	0.2533
M-0004	0.6222	0.5647	0.0125	0	0.0523	high risk	0.2666
M-0005	0.4444	0.1725	0.0012	0	0.0573	low risk	0.3333
M-0006	0.6222	0.4862	0.0126	0	0.0523	high risk	0.1733
M-0007	0.5333	0.3294	0.0126	0	0.0589	mid risk	0.1733
M-0008	0.1333	0.1725	0.0625	0.64	0.0656	high risk	0.3333
M-0009	0.4444	0.6431	0.0112	0	0.0523	mid risk	0.2933
M-0010	0.5333	0.4862	0.1500	0	0.0523	high risk	0.4266
M-0011	0.1777	0.1725	0.0126	0	0.0573	low risk	0.1733
M-0012	0.4444	0.4862	0.0125	0	0.0523	mid risk	0.1200
M-0013	0.3555	0.6274	0.0126	0	0.0581	low risk	0.2000
M-0014	0.4444	0.4078	0.0126	0.32	0.0523	mid risk	0.1333
M-0015	0.4444	0.4862	0.0625	0	0.0672	mid risk	0.5066

4.6. Multi-algorithm Performance

We divided the dataset into two parts: 90% training data and 10% testing data, with stratification and random states equaling 2. The performance of the NN, RF, and Naive Bayes algorithms is shown in table 8.

Table 8. Multi-algorithm performance

Algorithm	Accuracy	Precision	Recall	F1-Score
Neural Network	0.48	0.35	0.48	0.40
Random Forest	0.67	0.68	0.67	0.64
Naive Bayes	0.63	0.67	0.61	0.61

In table 8, the Random Forest algorithm achieved the highest accuracy (67%), followed by Naive Bayes (63%) and Neural Network (48%). Although Random Forest obtained slightly higher accuracy, this difference is relatively small and was not verified through statistical significance testing. Therefore, Random Forest can be described as performing marginally better rather than conclusively superior to Naive Bayes for this dataset. Future research should include statistical validation, such as cross-validation and McNemar's or paired t-tests, to determine whether these differences are significant. Although Random Forest achieved the highest accuracy among the tested algorithms, potential overfitting remains a concern. The model evaluation was based on a single 90/10 train–test split, which may not fully capture the model's generalization capability. Given that Random Forest can yield inflated accuracy on limited datasets, more rigorous validation techniques such as k-fold cross-validation, stratified sampling, or repeated hold-out testing should be applied in future studies. These methods would help ensure that model performance reflects true predictive power rather than overfitting to the training data. Additionally, hyperparameter optimization (e.g., number of trees, maximum depth, and feature selection per split) could further reduce overfitting risk and improve model robustness. The absence of confidence intervals or error margins in the reported metrics represents a limitation of this study. While the comparative accuracy and F1-scores suggest that Random Forest performs slightly better than Naive Bayes, the lack of statistical variability analysis limits the certainty of this conclusion. Incorporating repeated cross-validation or bootstrapped confidence intervals in future work would enhance the interpretability and robustness of the findings.

4.7. ROC Analysis

Receiver Operating Characteristic (ROC) curves were used to show and evaluate the performance of NN, RF, and Naive Bayes algorithms [26], [27]. Figure 4 showed ROC curves for RF, Naive Bayes, and NN, with the target risk classes, comprising "Low Risk," "Mid Risk," and "High Risk" represented in cyan is NN, orange is NB, as well as blue is RF.

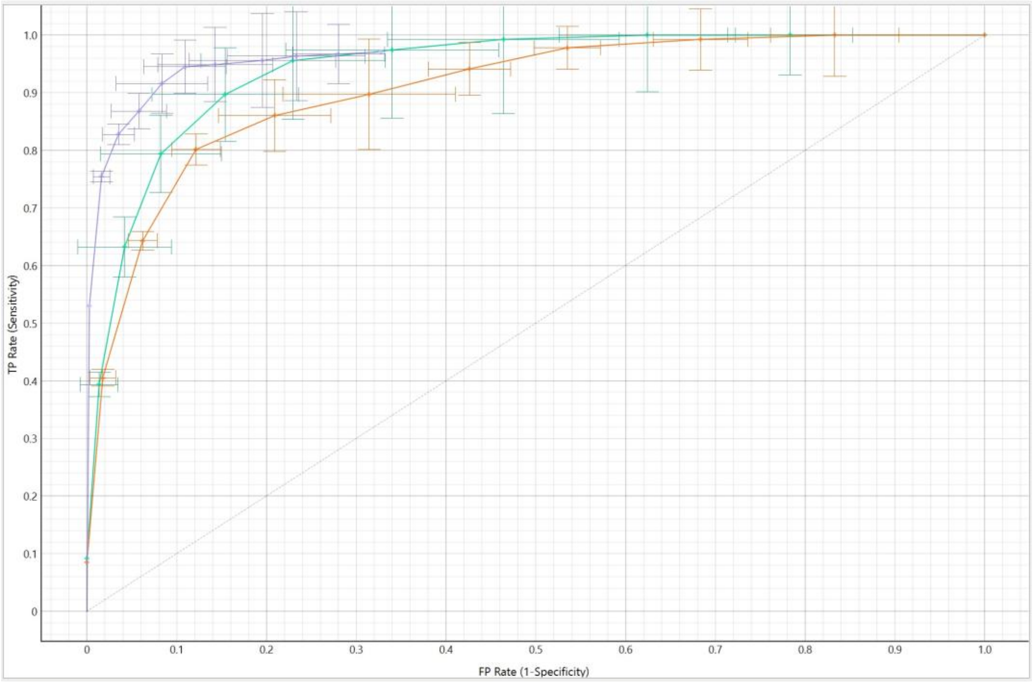

Figure 4. Performance curves of the NN, NB, and RF algorithms

4.8. Comparison of NB, NN, and RF Algorithms Prediction of Mater-nal and Child Health

Figure 5 showed a comparison of NN, Naive bayes, and RF algorithms for predicting maternal and child health risk.

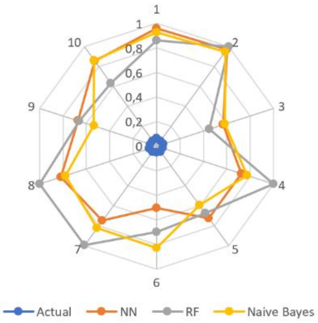

Figure 5. Results of the comparison of NN, NB, and RF algorithms on predicting maternal and child health risks

Table 9 presents the validation results comparing the actual risk level with predictions from the NN, Naive Bayes, and RF algorithms. In the digital version, color coding was originally used (green for correct predictions, red for mismatches). To ensure accessibility in grayscale or printed copies, we added alternative markers: an asterisk (*) indicates a mismatch between the predicted and actual classes, while a dash (–) denotes a correct match. For example, in the Naive Bayes column, five entries are marked with *, corresponding to cases where the predicted risk level differs from the actual outcome. This dual notation enables clear interpretation regardless of color or format.

Table 9. Validation results of predicting maternal and child health between NN, NB, and RF algorithms

1	2	3	4	5	6	7	8	9	10	11	12	12	14
M-0001	0.2000	0.5333	0.4862	0.1125	0.00	0.0656	high risk-	0.9614	high risk-	0.9246	high risk-	0.8611	high risk-
M-0002	0.3333	0.6222	0.6431	0.0875	0.00	0.0523	high risk-	0.9721	high risk-	0.9489	high risk-	1.0000	high risk-
M-0003	0.2533	0.1777	0.3294	0.0250	0.32	0.0606	high risk-	0.5676	mid risk*	0.5840	high risk-	0.4500	mid risk*
M-0004	0.2666	0.6222	0.5647	0.0125	0.00	0.0523	high risk-	0.7251	high risk-	0.7722	mid risk*	1.0000	high risk-
M-0005	0.3333	0.4444	0.1725	0.0012	0.00	0.0573	low risk-	0.7244	mid risk*	0.5916	mid risk*	0.6750	low risk-
M-0006	0.1733	0.6222	0.4862	0.0126	0.00	0.0523	high risk-	0.5028	mid risk*	0.8304	mid risk*	0.7000	high risk-
M-0007	0.1733	0.5333	0.3294	0.0126	0.00	0.0589	mid risk-	0.7488	mid risk-	0.8232	mid risk-	0.9966	mid risk-
M-0008	0.3333	0.1333	0.1725	0.0625	0.64	0.0656	high risk-	0.8170	high risk-	0.7800	high risk-	1.0000	high risk-
M-0009	0.2933	0.4444	0.6431	0.0112	0.00	0.0523	mid risk-	0.6726	low risk*	0.5361	mid risk-	0.6616	mid risk-
M-0010	0.4266	0.5333	0.4862	0.1500	0.00	0.0523	high risk-	0.8586	high risk-	0.8629	high risk-	0.6305	mid risk*

M-0011	0.1733	0.1777	0.1725	0.0126	0.00	0.0573	low risk-	0.7390	low risk-	0.7123	low risk-	0.9000	low risk-
M-0012	0.1200	0.4444	0.4862	0.0125	0.00	0.0523	mid risk-	0.5610	low risk*	0.5311	low risk*	0.7843	mid risk-
M-0013	0.2000	0.3555	0.6274	0.0126	0.00	0.0581	low risk-	0.6998	low risk-	0.4992	mid risk*	0.6250	low risk-
M-0014	0.1333	0.4444	0.4078	0.0126	0.32	0.0523	mid risk-	0.6692	mid risk-	0.5700	mid risk-	0.4750	mid risk-
M-0015	0,5066	0,4444	0,4862	0,0625	0.00	0.0672	mid risk-	0.6837	high risk-	0.7843	high risk-	0.6431	high risk-

Note: 1=Code patient, 2=Age, 3=Systolic Blood Pressure, 4=Diastolic Blood Pressure, 5=Blood Glucose, 6=Body Temperature, 7=Heart rate, 8=Risk level Status, 9=Neural Network Numerical, 10=Neural Network Validation Against Actual, 11=Naive Bayes Numerical, 12= Naive Bayes Validation Against Actual, 13= Random Forest Numerical, 14= Random Forest Validation Against Actual.

Table 9 showed the validation of maternal and child health risk predictions between the NN, Naive Bayes, and RF algorithms against the actual risk level results. The actual results were represented in green color in the "Risk level status" column. In addition, validation of NN algorithm was shown in the "NN validation against actual" column, where five data points were marked in red and differed from the "Risk level status" values. For the Naive Bayes algorithm validation in the "Validation of Naive Bayes against actual" column, there were also 5 data points that differed from the "Risk level status" and were marked in red. Meanwhile, the validation results of RF algorithm in the "Validation of RF against actual" column showed two data points that differed from the "Risk level status" and were marked in red.

The software developed in this study successfully predicts maternal and child health risks using the Naive Bayes algorithm and demonstrates its feasibility as a cost-effective decision-support tool. Comparative analysis with Random Forest and Neural Network models confirmed that the proposed approach achieves competitive accuracy while maintaining simplicity and accessibility for healthcare settings. However, several limitations should be acknowledged. The dataset was obtained from an open-source platform and may not fully represent diverse maternal populations or real-world clinical variability, limiting the model’s generalizability. Additionally, the relatively small sample size and lack of external validation restrict the robustness of the predictive outcomes. Ethical considerations must also be addressed before clinical implementation, particularly concerning data privacy, informed consent, and the potential consequences of algorithmic misclassification in medical decision-making. Future work should focus on validating the model using larger, clinically verified datasets and integrating safeguards to ensure ethical, transparent, and responsible AI deployment in maternal healthcare.

5. Conclusion

In conclusion, a software tool for predicting maternal and child health risks was developed using the Naive Bayes algorithm. The software's prediction accuracy was evaluated by comparing three algorithms: NN, RF, and Naive Bayes. The software analyzes seven input variables, including patient code, age, systolic and diastolic blood pressure, blood glucose, body temperature, and heart rate, to predict a single output. This output, known as the risk level status, is designed for use by pregnant women and healthcare providers. Testing on a maternal dataset showed that RF achieved 67% accuracy, followed by Naive Bayes at 63%, and NN at 48%, which yielded the lowest accuracy of the three. The total cost of developing the application is estimated at IDR 1,635,913.

6. Declarations

6.1. Author Contributions

Conceptualization: F.F., Z.Z., and S.R.; Methodology: Z.Z.; Software: F.F.; Validation: F.F., Z.Z., and S.R.; Formal Analysis: F.F., Z.Z., and S.R.; Investigation: F.F.; Resources: Z.Z.; Data Curation: Z.Z.; Writing Original Draft Preparation: F.F., Z.Z., and S.R.; Writing Review and Editing: Z.Z., F.F., and S.R.; Visualization: F.F.; All authors have read and agreed to the published version of the manuscript.

6.2. Data Availability Statement

The data presented in this study are available on request from the corresponding author.

6.3. Funding

The authors declare that this study was conducted without financial support from any governmental, commercial, or non-profit funding agency. No external funding was received for the execution of this research.

6.4. Institutional Review Board Statement

Not applicable.

6.5. Informed Consent Statement

Not applicable.

6.6. Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] E. D. Albrecht and G. J. Pepe, "Regulation of Uterine Spiral Artery Remodeling: a Review," *Reprod. Sci*., vol. 27, no. 10, pp. 1932–1942, 2020, doi: 10.1007/s43032-020-00212-8.

[2] A. Ornoy, M. Becker, L. Weinstein-Fudim, and Z. Ergaz, "Diabetes during pregnancy: A maternal disease complicating the course of pregnancy with long-term deleterious effects on the offspring. a clinical review," *Int. J. Mol. Sci*., vol. 22, no. 6, pp. 1–38, 2021, doi: 10.3390/ijms22062965.

[3] N. M. Hammoud, G. H. A. Visser, L. van Rossem, D. H. Biesma, J. M. Wit, and H. W. de Valk, "Long-term BMI and growth profiles in offspring of women with gestational diabetes," Diabetologia, vol. 61, no. 5, pp. 1037–1045, 2018, doi: 10.1007/s00125-018-4584-4.

[4] A. S. Desuky, S. Hussain, M. Akif Cifci, L. M. El Bakrawy, O. Mzoughi, and N. Kraiem, "Parameter Optimization Based Mud Ring Algorithm for Improving the Maternal Health Risk Prediction," *IEEE Access*, vol. 12, no. October, pp. 167245–167261, 2024, doi: 10.1109/ACCESS.2024.3495518.

[5] V. T. Siy Van, "Predicting undernutrition among elementary schoolchildren in the Philippines using machine learning algorithms," *Nutrition*, vol. 96, April 2022, p. 111571, 2022, doi: 10.1016/j.nut.2021.111571.

[6] R. G. Tiwari and A. Kumar, "Exploratory Data Analysis and Explainable Machine Learning for Classification of Maternal Health Risk Levels," *Proc. IEEE 2023 5th Int. Conf. Adv. Electron. Comput. Commun. ICAECC 2023*, June, 2024, pp. 1–8, 2023, doi: 10.1109/ICAECC59324.2023.10560341.

[7] Sudarmi, I. D. M. Sukrama, I. W. P. Sutirtayasa, I. W. Weta, and Irianto, "Influence of baby massage stimulation on the improvement of nutritional status, IGF-1, and cortisol level on undernourished infant," *Bali Med. J*., vol. 9, no. 1, pp. 36–40, 2020, doi: 10.15562/bmj.v9i1.1688.

[8] M. K. Açikgözoğlu, Ş. Pala, R. Atılgan, N. Ilhan, and N. Ilhan, "High serum angiopoietin-like protein-4 levels are associated with gestational hypertension and preeclampsia: a case-control study," *Turkish J. Biochem*., vol. 0, no. 0, pp. 2–6, 2024, doi: 10.1515/tjb-2023-0087.

[9] Rakefet Yoeli-Ullman, Howard J.A. Carp, and Shali Mazaki-Tovi, 3rd ed., CRC Press, 2020, pp. 1-12.

[10] E. Jung, "The etiology of preeclampsia," *Am. J. Obstet. Gynecol*., vol. 226, no. 2, pp. S844–S866, 2022, doi: 10.1016/j.ajog.2021.11.1356.

[11] G. Shaheen, "Association of endothelial nitric oxide synthase gene variants with preeclampsia," *Reprod. Health*, vol. 18, no. 1, pp. 1–15, 2021, doi: 10.1186/s12978-021-01213-9.

[12] K. Pavagada and J. Vemuri, "ScienceDirect ScienceDirect Prediction Prediction Of Of Maternal Maternal Health Health Risk Risk Factors Factors Using Using Machine Machine Learning Algorithms Learning Algorithms," *Procedia Comput. Sci*., vol. 258, no. 1, pp. 2713–2722, 2025, doi: 10.1016/j.procs.2025.04.532.

[13] S. Nahavandi, J. mine Seah, A. Shub, C. Houlihan, and E. I. Ekinci, "Biomarkers for macrosomia prediction in pregnancies affected by diabetes," *Front. Endocrinol. (Lausanne)*., vol. 9, no. July, pp. 1-12, 2018, doi: 10.3389/fendo.2018.00407.

[14] Z. Zulkifli, F. L. Gaol, A. Trisetyarso, and W. Budiharto, "Multi-algorithm for predicting the level accuracy of fault output in software," *Sci. Eng. Heal. Stud*., vol. 18, no. 1, pp. 1–16, Dec 2024, doi: 10.69598/sehs.18.24020006.

[15] R. Gulakala, B. Markert, and M. Stoffel, "Rapid diagnosis of Covid-19 infections by a progressively growing GAN and CNN optimisation," *Comput. Methods Programs Biomed*., vol. 229, no. 1, pp. 1-12, February 2023, doi: 10.1016/j.cmpb.2022.107262.

[16] A. Z. AKTAŞ, E. YAĞDERELİ, and D. SERDAROĞLU, "An Introduction to Software Testing Methodologies," *Gazi Univ. J. Sci. Part A Eng. Innov.,* vol. 8, no. 1, pp. 1–15, 2021.

[17] Z. Zulkifli, F. A. Makkiyah, D. Antoni, F. Fitriana, T. Jamaan, and A. Taufik, "Multi-Algorithm to Measure the Accuracy Level of Diabetes Status Prediction," *J. Appl. Data Sci*., vol. 5, no. 2, pp. 736–746, 2024, doi: 10.47738/jads.v5i2.250.

[18] N. Ghaffar Nia, E. Kaplanoglu, and A. Nasab, "Evaluation of artificial intelligence techniques in disease diagnosis and prediction," *Discov. Artif. Intell*., vol. 3, no. 1, Art. no. 5304, pp. 1-12, 2023, doi: 10.1007/s44163-023-00049-5.

[19] L. Jamel, "Improving prediction of maternal health risks using PCA features and TreeNet model," *PeerJ Comput. Sci*., vol. 10, no. 1, pp. 1–24, 2024, doi: 10.7717/peerj-cs.1982.

[20] D. Estrela, "Molecular Biomarkers for Timely and Personalized Prediction of Maternal-Fetal Health Risk," *Biomolecules*, vol. 15, no. 3, pp. 1–15, 2025, doi: 10.3390/biom15030312.

[21] S. S. Cross, R. F. Harrison, and R. L. Kennedy, "Introduction to neural networks", *lancet,* vol. 346, no. 8982. 1995, pp. 5 doi: 10.1016/S0140-6736(95)91746-2.

[22] E. Dritsas and M. Trigka, "Data-Driven Machine-Learning Methods for Diabetes Risk Prediction," *Sensors*, vol. 22, no. 14, pp. 18 2022, doi: 10.3390/s22145304.

[23] S. A. Ali, N. R. Roy, and D. Raj, "Software Defect Prediction using Machine Learning," Proc. 17th INDIACom; 2023 10th Int. Conf. Comput. Sustain. Glob. Dev. INDIACom 2023, vol. 2023, no. Icoei, pp. 639–642, 2023.

[24] S. Nurdin and D. Hamdani, "The Design of the Building Rental Information System Uses the Prototype Method," *IJISCS (International J. Inf. Syst. Comput. Sci*., vol. 5, no. 2, pp. 119–131, 2021, doi: https://doi.org/10.56327/ijiscs.v5i2.1022.

[25] S. Mehta et al., "Simplify-then-translate: Automatic preprocessing for black-box translation," *AAAI 2020 - 34th AAAI Conf. Artif. Intell*., vol. 34, no. 05, pp. 8488–8495, 2020, doi: 10.1609/aaai.v34i05.6369.

[26] O. A. Lazzarini Lemos, F. Fagundes Silveira, F. Cutigi Ferrari, and A. Garcia, "The impact of Software Testing education on code reliability: An empirical assessment," *J. Syst. Softw*., vol. 137, no. mar., pp. 497–511, March 2018, doi: 10.1016/j.jss.2017.02.042.

[27] W. Aiken, H. Kim, S. Woo, and J. Ryoo, "Neural network laundering: Removing black-box backdoor watermarks from deep neural networks," *Comput. Secur*., vol. 106, no. July, pp. 1-12, 2021.

[28] Z. Zulkifli, M. Mardiana, and D. Despa, "Software Quality Assessment Model: A New Approach for Software Testing Tools," *Int. J. Softw. Eng. Knowl. Eng*., vol. 35, no. 2, pp. 139–155, 2024, doi: 10.1142/S0218194024500517.

[29] O. Esteban, "fMRIPrep: a robust preprocessing pipeline for functional MRI," *Nat. Methods*, vol. 16, no. 1, pp. 111–116, 2019, doi: 10.1038/s41592-018-0235-4.

[30] T. Byun, V. Sharma, A. Vijayakumar, S. Rayadurgam, and D. Cofer, "Input Prioritization for Testing Neural Networks," in *2019 IEEE International Conference On Artificial Intelligence Testing (AITest)*, vol. 2019, no. may, pp. 63–70, 2019. doi: 10.1109/AITest.2019.000-6.