

Expert Web -Based System for Computer Troubleshooting Diagnosis Using Forward Chaining Method

Rayhand Ferdiano¹, Ami Anggraini Samudra^{*2}, Rahayu Trisetyowati Untari³

^{1,2,3} Informatics Education Study Program, Faculty of Science and Technology, Universitas PGRI Sumatera Barat, Padang, 25173, Indonesia

amianggrainisamudra@gmail.com

Abstract

Computer failure diagnosis at SMKN 2 Padang Panjang is still done manually and relies on a single technician, resulting in lengthy identification and repair processes. Furthermore, the recording of computer failure history is not organized, making it difficult to track failure patterns in specific units. This research aims to develop an *Expert A web* -based system with the *Forward Chaining method* to assist the computer *troubleshooting diagnosis process* based on symptoms reported by users. The type of research used is *Research and Development (R&D)* with a *Rapid Application Development (RAD)* development model consisting of the *Requirements Planning, User Design, Construction, and Cutover stages*. The knowledge base (*rule base*) is formed automatically from training data using the *Decision Tree C4.5 algorithm*, then traced using the *Forward Chaining inference engine*. The system is equipped with a *QR Code* feature as a unique identity for each computer as well as a diagnosis history feature. System testing is carried out through *Black Box Testing, White Box Testing*, expert evaluation, and user testing. The results of *black box* and *white box testing* show that all system functions run according to design. Expert evaluation obtained an average score of 95.8% with a very valid category, while testing on 25 users showed a level of conformity between the system diagnosis and the expert diagnosis of 96%. These results prove that *Expert The web* -based system developed is effective, valid, and suitable for use as an initial diagnostic tool for computer *troubleshooting* in school laboratory environments.

Keywords: *Expert system, Forward Chaining, decision tree c4.5, computer troubleshooting, qr code*

JPTI Jurnal is licensed under a Creative Commons Attribution-Share Alike 4.0 International License.



1. Introduction

The rapid advancement of information technology in the era of the Industrial Revolution 4.0 has transformed the educational paradigm, particularly in the vocational field. Computers have become a vital tool in supporting the learning process, both theoretically and practically, making mastery of computer technical competencies a primary requirement for vocational students facing the digital-based workplace.

SMK Negeri 2 Padang Panjang utilizes computers scattered throughout various rooms, such as the TKJ, MLD, and RPL laboratories, the teachers' lounge, the administration room, and the library. The high intensity of computer use increases the likelihood of damage, both to hardware and software. A frequent problem is the lengthy repair time due to the limited number of technicians with expertise in this field, while the diagnostic process still relies heavily on the technician's experience, thus tending to be time-consuming. Recording of computer damage history is also still done manually, resulting in a lack of organization and making it difficult to track recurring damage on a particular computer. On the other hand, students still have difficulty troubleshooting independently because there is no systematic diagnostic guide available, so some of the lab time is often wasted waiting for the repair process.

Previous research supports an expert system-based approach to address this issue. Farizi [1] concluded that an expert system with the *Forward Chaining method* facilitates the diagnostic process without having to rely on books or the presence of experts in person. Darmawan et al. [2] also stated that expert system applications make it easier for technicians to diagnose computer damage more efficiently. With the *Forward Chaining method*, expert systems can automatically analyze user-reported symptoms and provide relevant solutions, thereby accelerating the identification and handling of technical problems.

Based on these problems, this research developed *Expert A web* -based system for computer *troubleshooting diagnostics* at SMKN 2 Padang Panjang. The system's novelty lies not only in the diagnostic process, but also in the ability to automatically generate a knowledge base from training data using the *Decision Tree C4.5 algorithm*, so that adding or changing data does not require manual *rule creation*. Each computer unit is also equipped with a *QR Code* as a unique identity to access the diagnosis page and record inspection history, so that admins or technicians can monitor frequent damage patterns on each unit for evaluation and maintenance purposes.

Based on the description above, the formulation of the research problem is how to develop *Expert A web* -based system using the *Forward Chaining method* to assist the

computer *troubleshooting diagnostic process* at SMKN 2 Padang Panjang. The aim of this research is to produce an *Expert System*. A *web* -based system that applies the *Forward Chaining method* to help technicians and students identify the type of computer damage based on the symptoms that appear, provide initial handling solutions, and support the *troubleshooting process* in the computer laboratory to be faster, more effective, and more efficient.

2. Research Methods

Research and Development (R&D) method, a research method that aims to produce a product while simultaneously testing its feasibility. The product developed is an *Expert A web* -based system to assist the computer *troubleshooting diagnostic process* in the TKJ, MLD, and RPL Laboratories of SMKN 2 Padang Panjang using the *Forward Chaining method*.

2.1 Development Model

The development model used is *Rapid Application Development (RAD)*, one of the *System Development models*. *Development Life Cycle (SDLC)* which emphasizes short, flexible development cycles and active user involvement [3]. The RAD model consists of the following four stages.

Requirements Planning is the stage of identifying system requirements through observation and interviews with laboratory technicians to identify common *troubleshooting issues*. *Knowledge acquisition* is conducted through *literature review*, *expert interviews*, and *direct observation of computer diagnostic and repair processes*. *Data on symptoms, types of damage, causes, and solutions* are then compiled into a training dataset.

User Design is the stage of designing the system flow, diagnostic process, knowledge base, database, and interface. The training dataset is processed using the *Decision Tree C4.5* algorithm to form a decision tree, which is then converted into a *rule. IF-THEN* as a knowledge base. The *Forward Chaining* inference engine searches these *rules* based on user-selected facts or symptoms to arrive at a conclusion regarding the type of damage and its solution.

Construction is the stage of implementing the design into a system using the *Python programming language* with the *Flask framework* and *SQLite* database. At this stage, features for managing symptom data, damage, solutions, and *rules* are built; a diagnostic process using *Forward Chaining*; diagnostic history storage; a *QR Code* feature; and user and admin interfaces are built.

Cutover is the testing stage of the system that has been built, including *Black Box Testing*, *White Box Testing*, expert evaluation, and user testing, to ensure that all functions run as needed before the system is implemented.

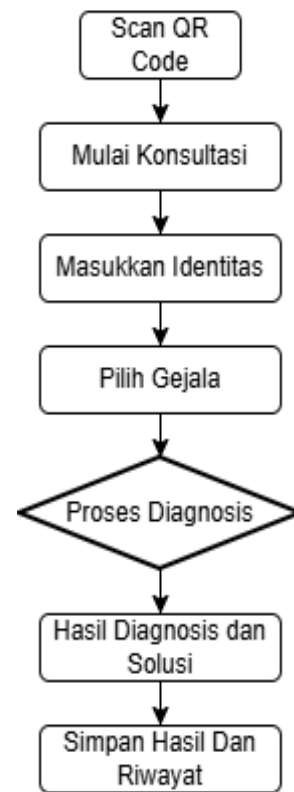


Figure 1. Diagnosis Process Flow in Expert System

2.2 Research Subjects and Instruments

The research subjects consisted of 25 students of class XI TKJ as users, 1 laboratory technician, and 2 Informatics Education lecturers as system experts, with a total of 28 respondents. The research instrument consisted of a *checklist sheet (test case)* for *Black box* and *White box testing* at the Alpha Testing stage, as well as a four-level *Likert scale questionnaire* referring to the ISO 25010 standard for *Beta Testing* , which includes system functional indicators, ease of use, interface display, navigation, clarity of information, system performance, feature integration, reliability, and system feasibility.

Subject	Role	Amount
Class XI TKJ students	User	25
TKJ Labor Technician	System Expert	1
Informatics Education Lecturer	System Expert	2
Total		28

Table 1. Research Subjects

2.3 Data Analysis Techniques

Black Box Testing data was analyzed descriptively by comparing actual results to expected results in each test scenario. *Alpha* and *Beta Testing data* were analyzed using the percentage of scores to the maximum score, using the following formula.

$$\text{Value (\%)} = (\text{Score Obtained} / \text{Maximum Score}) \times 100\% \quad (1)$$

The *Alpha Testing* percentage results are interpreted as very valid (81%-100%), valid (61%-80%), less valid (41%-60%), and invalid (0%-40%). The system is declared worthy of continuing to *Beta Testing* if the *Alpha*

Testing results are at least in the valid category ($\geq 61\%$) [4]. The effectiveness of the system is analyzed by comparing the results of the system diagnosis to the expert diagnosis as a reference, using the same percentage formula; the system is declared effective if it obtains a value of $\geq 80\%$.

3. Results and Discussion

3.1 System Implementation

Expert The web-based system was developed with the Python programming language, the Flask framework, and the SQLite database. The system provides two types of users, namely admin/technician and User. Admin can manage computer data, symptoms, damage, solutions, rule bases, user data, QR Codes, and view the history of diagnosis results. Users access the system by scanning the QR Code on the computer unit, entering their identity, then answering the symptom questions step by step with the options "Yes" or "No". Each answer is used as a fact in the rule tracing process by the Forward Chaining inference engine until a conclusion is obtained in the form of the type of damage and handling solution, which is then displayed on the diagnosis results page and saved as a history.



Figure 2. Display of Diagnosis Results on the User Side

Figure 2 shows the diagnostic results page the user receives after the symptom tracing process is complete. The system displays the identity of the room/computer being scanned, the type of damage detected (e.g., hard disk/SSD damage due to *bad sectors*), and recommended initial handling solutions the user can implement immediately. The results are automatically saved to a SQLite database.

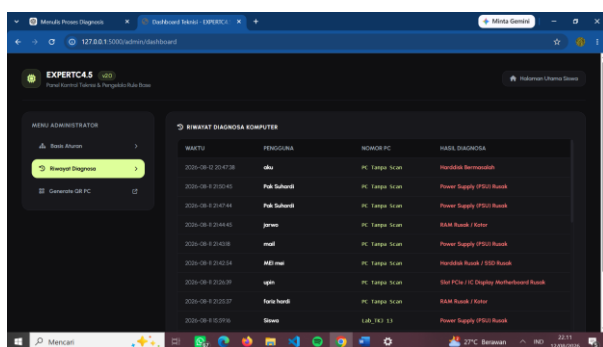


Figure 3. Diagnosis History on the Admin/Technician Dashboard

Figure 3 shows the diagnosis history page on the admin/technician dashboard, which displays the diagnosis time, username, computer ID, and diagnosis results for each test. This feature allows admins/technicians to monitor recurring failure patterns on specific units as a basis for evaluating and maintaining laboratory computers.

3.2 Black Box and White Box Test Results

Black box testing was conducted on the admin login function, filling in user identity, diagnosis process, symptom answer selection, display of diagnosis results, history storage, rule base management, and QR Code creation. All test scenarios showed actual results in accordance with the expected results, so that all system functions were declared to be running well. *White box* testing focused on the logic of the rule formation process with Decision Tree C4.5 and the Forward Chaining inference process; the results of the execution path tracing showed that all program paths could be executed without logical errors, so that the implementation was in accordance with the system design.

3.3 Expert Evaluation Results

The expert evaluation was carried out by three validators (one laboratory technician and two Informatics Education lecturers) using a questionnaire based on the ISO 25010 standard, as presented in Table 2.

Assessment Aspects	Average (%)	Category
System functionality	95.9	Very Valid
Ease of use	91.7	Very Valid
Interface view	91.7	Very Valid
System navigation	100	Very Valid
Clarity of information	91.7	Very Valid
System performance	100	Very Valid
Feature integration	100	Very Valid
System reliability	91.7	Very Valid
System feasibility	100	Very Valid
Overall average	95.8	Very Valid

Table 2. Expert Evaluation Results

Based on Table 2, the expert evaluation obtained an average of 95.8% with a very valid category, which indicates that the system workflow, both functionally and non-functionally, has met the eligibility criteria for use as a diagnostic tool at SMKN 2 Padang Panjang.

3.4 User Testing Results

User testing involved 24 TKJ students and one TKJ teacher who used the system directly in a computer laboratory. The summary results of the user questionnaire are shown in Table 3.

Assessment Aspects	Mark (%)	Category
System functionality	85.0	Very Valid
Ease of use	86.5	Very Valid
Interface view	89.5	Very Valid
System navigation	88.0	Very Valid
Clarity of information	89.0	Very Valid

Assessment Aspects	Mark (%)	Category
System performance	89.0	Very Valid
Feature integration	90.0	Very Valid
System feasibility	90.0	Very Valid
Overall average	88.3	Very Valid

Table 3. User Testing Results

In addition to user perception assessments, the effectiveness of the diagnostic results was tested by comparing the system's output to expert diagnoses. Of the 25 diagnostic processes tested, 24 produced conclusions that matched the actual damage conditions, resulting in a diagnostic agreement rate of 96%. This value is above the established effectiveness threshold ($\geq 80\%$), thus the system is considered effective as a diagnostic tool for computer *troubleshooting*.

3.5 Discussion

Black box and *White box* testing show that all the main functions of the system run according to design, in line with Rafli et al. [5] who stated that *White Box Testing* is effectively used to check the correctness of the program's logic flow through flowcharts and cyclomatic complexity calculations. This finding is also supported by Pasiak [6], who shows that the application of a computer damage diagnosis expert system makes it easier for users to identify problems that occur on their devices.

The expert evaluation score of 95.8% indicates that the system is functionally and non-functionally suitable for submission to the user testing phase. The 96% diagnostic suitability level in user testing demonstrates that the integration of the *Decision Tree* C4.5 algorithm for *rule formation and the Forward Chaining* method as an inference engine is capable of producing diagnoses that are close to expert decisions. Thus, *Expert This web -based system* can be used to help speed up the process of early identification of computer damage, reduce dependence on a single technician, and provide a structured diagnostic history as material for evaluating computer maintenance in school laboratories.

4. Conclusion

Expert A web -based system for computer *troubleshooting diagnosis* at SMKN 2 Padang Panjang was successfully developed using the *Rapid Application Development* (RAD) model with a knowledge base formed automatically through the *Decision Tree* C4.5 algorithm and traced using the *Forward Chaining method*, equipped with *QR Code features* and diagnosis history. The results of *Black box* and *White box testing* showed that all system functions ran according to design. Expert evaluation obtained a score of 95.8% (very valid), while testing of 25 cases of diagnosis by users resulted in a 96% agreement level with expert diagnoses. These results prove that the developed system is valid, practical, and effective for use as an initial diagnosis tool for computer *troubleshooting*, while supporting the management of computer damage information in a more focused and systematic manner in the school laboratory. Further research is recommended to expand the knowledge base by adding symptoms and types of damage, involving more experts in *rule development*,

and testing the system on a more diverse number of cases and laboratory environments.

Thank-you note

The author would like to thank the Informatics Education Study Program, Faculty of Science and Technology, Universitas PGRI West Sumatra, and SMK Negeri 2 Padang Panjang for their support and permission during the research. The author declares that there is no conflict of interest in this research and that he has not received any specific grants from any funding agency.

Bibliography

- [1] A. Farizi, "Expert System for Diagnosing Computer Damage Using the *Forward Chaining Method* ," Edu Komputika Journal, vol. 1, no. 2, pp. 21–32, 2021.
- [2] R. Darmawan, R. Darni, UN Padang, and K. Padang, "Using the *Forward Chaining Method* in the Multimedia Laboratory of State Universities," Jurnal Media Akademik (JMA), vol. 3, no. 6, 2025.
- [3] Nurman Hidayat and Kusuma Hati, "Application of the *Rapid Application Development* (RAD) Method in the Design of the Online Report Information System (SIRALINE)," Information Systems Journal, vol. 10, no. 1, pp. 8–17, 2021, doi: 10.51998/jsi.v10i1.352.
- [4] HL Hakim, D. Faqih, D. Deva, IF Hudaya, and MN Ilyas, "Alpha and Beta Testing on TIJE Application," TeknoIS: Scientific Journal of Information Technology and Science, vol. 14, no. 2, pp. 285–295, 2024, doi: 10.36350/jbs.v14i2.265.
- [5] H. Rafli, R. Zen, and I. Nuryasin, "Application of Whitebox Testing in System Testing Using Basis Path Technique," Journal of Information Systems and Informatics Engineering, vol. 8, no. 1, pp. 101–111, 2024, doi: 10.35145/joisie.v8i1.4229.
- [6] DISMAG Charles Saputra Pasiak, "Computers Using the *Forward Chaining Method* , Sriwijaya State Polytechnic," pp. 1–6, 2022.
- [7] O. Saputra, I. Fitri, and ET Esti Handayani, "Expert System for Diagnosing Computer *Hardware Damage* Using the *Forward Chaining* and Certainty Factor Methods Based on a Website ," Jurnal JTik, vol. 6, no. 2, pp. 234–242, 2022, doi: 10.35870/jtik.v6i2.416.
- [8] A. Hasnan and A. Willy, "Expert System for Diagnosing Laptop *Hardware Damage* Using Web- Based *Forward Chaining Method* ," Jurnal Surya Informatika, vol. 12, no. 2, pp. 1–7, 2022, doi: 10.48144/surya i nformatika.v12i2.1364.
- [9] J. Kusuma, A. Jinan, and Z. Situmorang, "Application of *Decision Tree* Algorithm C4.5 in Determining Shiploading License," MEANS, vol. 7, no. 1, pp. 72–76, 2022.
- [10] R. Sidik and K. Aryansyah, "Implementation of *QR Codes* in the Development of Workshop and Seminar Attendance Information Systems," Journal of Information Management (JAMIKA), vol. 11, no. 2, pp. 88–101, 2021, doi: 10.34010/jamika.v11i2.4676.
- [11] DF Ningtyas and N. Setiyawati, "Implementation of Flask Framework in Purchasing Approval Request Application Development," Jurnal Janitra Informatika dan Sistem Informasi, vol. 1, no. 1, pp. 19–34, 2021, doi: 10.25008/janitra.v1i1.120.
- [12] FA Nugraha and YA Susetyo, "Comparative Analysis of DuckDB and *SQLite Database Performance* in Big Data Processing," JIPI, vol. 8, no. 3, pp. 1052–1060, 2023, doi: 10.29100/jipi.v8i3.4032.