KOMTEKINFO JOURNAL
LPPM Universitas Putra Indonesia YPTK Padang
Jl. Raya Lubuk Begalung, Padang, West Sumatera, Indonesia, Zip Code: 25221
Volume: 13, Issue: 1, Page: 1 - 10, March 30th, 2026, e-ISSN: 2502-8758
Available online at website: https://jkomtekinfo.org/ojs/index.php/komtekinfo

Analysis of The Feasibility Level of IT Device Using K-Means Cluster and C4.5 Classification

Fachriqi Naldes[1]✉, S. Sumijan[2], Syafri Arlis[3]
[1,2,3] Master of Informatics Engineering, Faculty of Computer Science, Universitas Putra Indonesia YPTK, Padang, 25221, Indonesia
fachriqinaldes@gmail.com

Abstract

The availability of reliable laptops is essential for ensuring smooth business operations; however, decisions regarding device upgrades and replacements in many organizations still rely primarily on device age and subjective user perceptions. This practice often leads to inconsistent IT asset lifecycle decisions, increased security risks, and inefficient cost management. This study proposes a classification model to recommend laptop feasibility levels, namely usable, requires upgrade, and requires replacement, based on a combination of technical specifications and operating system characteristics. K-Means clustering is applied to group laptops into three feasibility categories using processor type, release year, RAM capacity, storage type, and operating system attributes that have undergone performance score–based ordinal encoding and Min–Max normalization. Subsequently, the C4.5 algorithm is employed to construct a decision tree using the K-Means cluster labels as target classes, producing interpretable if–then rules that describe device feasibility patterns. The dataset is obtained from the IT device inventory of PT Semen Indonesia, consisting of 1,905 laptop records, which after data cleaning result in 85 unique specification combinations for analysis. The clustering process classifies 47 laptops as usable, 22 as requiring upgrades, and 16 as requiring replacement. The C4.5 algorithm model achieves accuracy, precision, recall, and F1-score values of 100% on the test data, indicating its ability to effectively replicate the feasibility patterns generated by K-Means algorithm. These findings demonstrate that the proposed approach provides a data-driven framework for supporting upgrade and replacement decisions, contributing to more efficient and measurable IT asset lifecycle management.

Keywords: feasibility level, IT device, K-Means cluster, C4.5 classification

1. Introduction

Large-scale organizations rely on the availability and performance of laptops as a core component of IT infrastructure to maintain operations. Asset lifecycle management provides a framework from planning, procurement, and usage to maintenance and replacement allowing decision-makers to assess risks and opportunities at each phase in a measurable way. Recent literature emphasizes that a lifecycle perspective maps economic, social, and environmental risks at every stage of an asset, ensuring that hardware renewal and rejuvenation decisions are more targeted [1].

Over time, devices with aging specifications and outdated software support tend to increase security risks, reduce software compatibility, and add to the maintenance burden. Operating system vulnerability studies map trends and the severity of weaknesses across Windows, Linux, and macOS platforms, highlighting the need for regular updates and disciplined device refresh plans. These facts reinforce the urgency of device refresh policies and the establishment of IT device feasibility standards in large-scale work environments [2].

Sustainability policies demand a balance between device replacement and upgrades based on circular economy principles. Recent reviews indicate that machine learning assists in operation scheduling, quality control, and workflows within the context of data-driven asset remanufacturing and rejuvenation. This framework justifies the integration of analytics when organizations simultaneously weigh productivity, reliability, security, cost, and environmental impact to determine whether a device remains fit for use, requires an upgrade, or needs replacement [3-4].

Previous research in operational contexts demonstrates the role of clustering in mapping entities into action groups that are easily understood by stakeholders. A 2024 JTIIK article reported a K-Means algorithm implementation that produced three indicator-based clusters, providing clear pattern recognition for tactical decisions. Similarly, a 2023 RESTI article applied K-Means algorithm to project health monitoring and presented status mapping as a basis for prioritizing actions [5-6].

The concept of Knowledge Discovery in Databases (KDD) provides an overarching framework that positions data mining as one stage in the knowledge

Submitted: 06-01-2026 | Revised: 01-02-2026 | Accepted: 04-03-2026 | Published: 30-03-2026

discovery process. The KDD process includes data selection, cleaning, transformation, data mining, and knowledge evaluation and interpretation, meaning the quality of the results is heavily influenced by discipline in the early stages. Modern KDD methodologies demonstrate systematic process designs, ensuring that the generated knowledge is ready for execution across various applied data fields, including IT asset management [7-8]

Data mining acts as a direct derivative of KDD, focusing on techniques to extract patterns from structured and semi-structured data. Current clustering taxonomies position K-Means as a simple, fast, and scalable partitional approach suitable for initial segmentation across various fields. Recent decision tree reviews summarize the strengths of interpretability, pruning practices, and model limitations in the context of auditable decision-making, making them relevant for forming decision rules for IT device feasibility [9-10].

Methods aligned with operational decision needs in organizations include K-Means for clustering and C4.5 for classification. K-Means minimizes intra-cluster distance and maximizes inter-cluster separation in the feature space, allowing the cluster map to be interpreted as action segments such as devices that are fit for use, require upgrades, or need replacement. C4.5 derives concise and auditable "if-then" rules, enabling organizations to translate analytical results into operational Standard Operating Procedures (SOPs) for device feasibility [9-10].

Previous research using these methods demonstrates their relevance in applied contexts. A 2024 RESTI article improved C4.5 performance using Particle Swarm Optimization in the healthcare sector, increasing classification accuracy while maintaining readability. A 2024 JTIIK article showed the success of K-Means in generating three representative clusters from operational data as a basis for policy intervention, indicating that the combination of clustering and classification methods has significant potential for IT asset management [11], [5].

The operational link between concepts and methods in problem-solving is presented in the KDD workflow, which positions K-Means for initial action-group mapping and C4.5 for forming decision rules. Cluster maps provide a common language for stakeholders, while if-then rules translate patterns into consistent, auditable operational guidelines. The integration of these concepts and methods offers a transparent path from data to policy decisions regarding the feasibility of IT devices, specifically laptops, within an organizational environment [7], [10].

Consequently, this research addresses the formulation of laptop feasibility decisions at PT Semen Indonesia. Laptop feasibility holds a strategic position in ensuring business process continuity, information security, and operational cost efficiency; thus, comprehensive analysis is required to support IT asset management decisions. Applying data mining techniques through K-Means and C4.5 algorithms provides a systematic way to categorize devices into different condition segments (fit for use, upgrade required, replacement required) while deriving clear decision rules. This allows for the identification of key factors determining feasibility to serve as a policy foundation. This study aims to provide objective, consistent, and auditable decisions based on laptop specification data to support more effective asset lifecycle management within the company.

2. Methods

The design of this research framework is based on the general principles of the Knowledge Discovery in Databases (KDD) process and is specifically adapted to the context of IT asset management at PT Semen Indonesia. The research begins with conceptual stages, including problem identification and a literature review, which serve as the foundation for the analytical approach.

Subsequently, the study proceeds to technical stages comprising data extraction, data preprocessing, and data analysis using data mining algorithms. The final stages involve model evaluation, result interpretation, and implementation, followed by systematic documentation of findings and discussion. This structured workflow is intended to ensure that the research process is clear, logical, and reproducible. An overview of the research framework and its stages is presented in Figure 2.1.

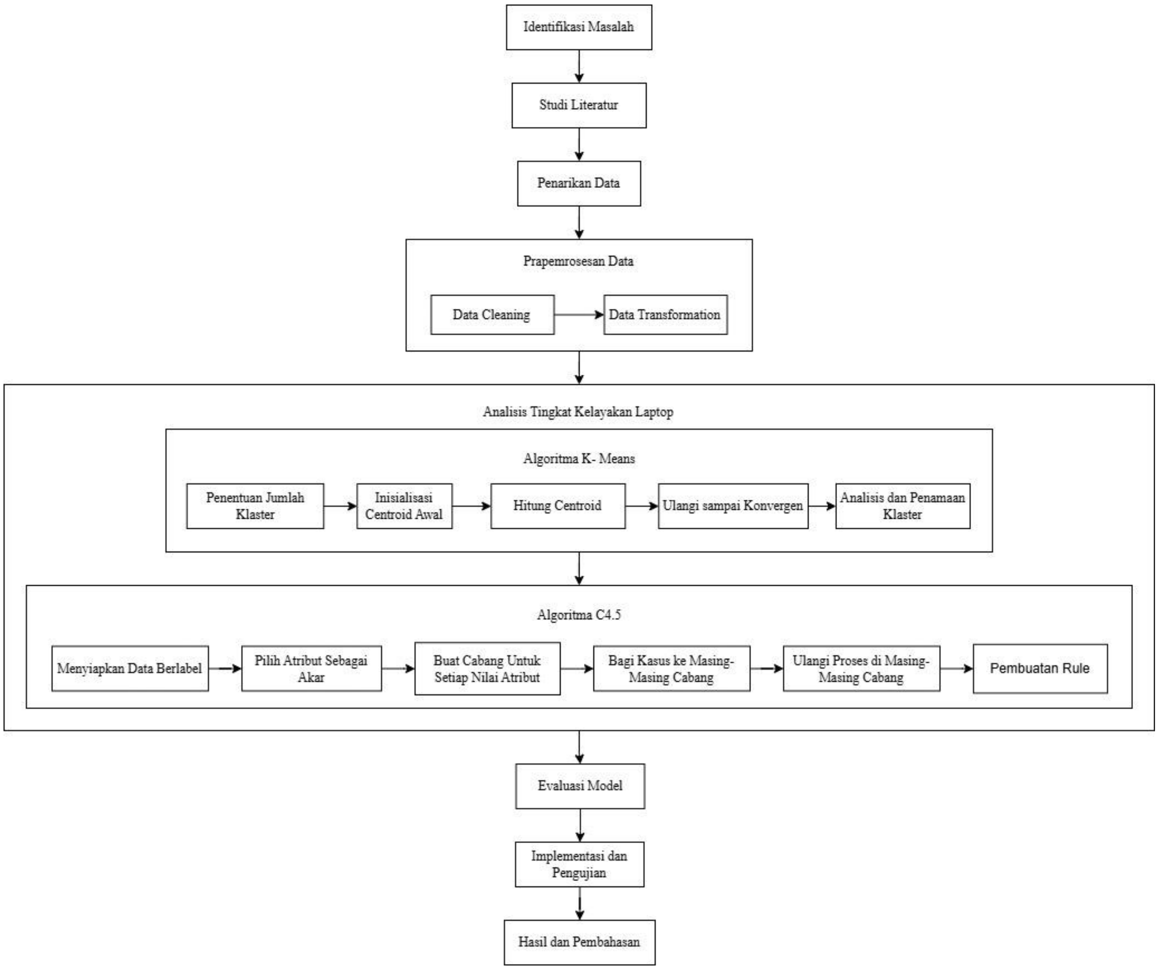

Figure 2.1. Research Framework

2.1 Preprocessing

Data preprocessing was conducted to prepare the dataset prior to analysis using the K-Means and C4.5 algorithms. This stage aimed to ensure data quality, consistency, and compatibility with analytical algorithms, as emphasized in the Knowledge Discovery in Databases (KDD) process [12-13]. Previous studies have demonstrated that effective preprocessing directly affects the accuracy, stability, and generalization ability of machine learning models, while inadequate preprocessing may lead to biased and unreliable results [14-15]. In this study, data preprocessing consisted of two main steps: data cleaning and data transformation.

Data cleaning was performed to eliminate irrelevant, incomplete, and inconsistent records. Only attributes directly related to the technical characteristics of laptops were retained, namely processor type, release year, RAM capacity, storage type, and operating system. Records containing missing values were removed to ensure data completeness. In addition, textual attributes were standardized to address writing inconsistencies and prevent semantically identical values from being treated as different categories. Duplicate records were identified and removed so that each entry represented a unique laptop configuration. These procedures are consistent with prior findings that systematic data cleaning is essential for producing reliable and generalizable data mining results [16], [14], [17].

Data transformation was then applied to convert all attributes into numerical representations with comparable scales. Categorical attributes were encoded using an ordinal scoring approach that reflects relative performance or technological maturity, which has been shown to be more informative than purely nominal encoding for attributes with inherent order [13]. The resulting scores were normalized to the range [0,1] using Min–Max scaling. Numerical attributes, namely release year and RAM capacity, were normalized using the same technique to prevent scale dominance in distance-based calculations. Previous studies highlight that appropriate scaling is particularly important for algorithms such as K-Means, which rely on distance measures and are sensitive to differences in feature ranges [18-19], [15].

Through these preprocessing steps, the dataset was transformed into a consistent numerical form that supports proportional feature contributions during clustering and classification, thereby ensuring that the application of the K-Means and C4.5 algorithms yields reliable and reproducible results.

2.2 K-Means Algorithm

K-Means is a partitional clustering algorithm that partitions a dataset into K clusters by minimizing the within-cluster sum of squares (WCSS), thereby producing clusters that are internally compact and externally well separated [20], [9]. Given a set of numerical feature vectors, K-Means iteratively assigns objects to the nearest centroid and updates centroid positions until convergence.

The performance of K-Means is influenced by centroid initialization, feature scaling, and the choice of K. Prior studies report that improper scaling may degrade clustering quality, whereas normalization or standardization improves cluster compactness and stability, particularly for distance-based algorithms such as K-Means [18], [21].

The general stages of the K-Means algorithm applied in this study are as follows.

1. Determine the number of clusters K
2. Initialize centroids.
3. Calculate the distance of each object and assign the object to the nearest cluster (assignment step), which is calculated using:

$$\parallel x - y \parallel_2 = \sqrt{\sum_{l=1}^{d} (x_l - y_l)^2} \tag{1}$$

In this equation, x represents the first data vector with d dimensions (attributes), while y represents the second data vector of the same dimension, typically representing the cluster center. The component x_l denotes the value of the l-th attribute of vector x, and y_l denotes the value of the l-th attribute of vector y. The value d indicates the total number of attributes or features used in the distance calculation. The resulting Euclidean distance serves as the basis for assigning each data object to the cluster with the smallest distance.

4. Update the centroids and repeat the process until the convergence condition is met, calculated using:

$$\mu_i^{(new)} = \frac{1}{|C_i|} \sum_{x_j \in C_i} x_j \tag{2}$$

In this equation, μ_i^(new) represents the centroid of the i-th cluster after the update process. The symbol C_i denotes the set of data points that belong to the i-th cluster, while $|C_i|$ indicates the total number of data points within that cluster. The variable x_j refers to a data vector that is a member of the cluster C_i. This update process computes the new centroid as the average of all data points assigned to the corresponding cluster.

5. Analysis and labeling of clusters based on the characteristics of each cluster derived from centroid values and the distribution of their members. Based on this analysis, each cluster is assigned a label (usable, requires upgrade, and requires replacement), enabling the K-Means results to be more easily interpreted and utilized for decision-making purposes.

2.3 C4.5 Algorithm

C4.5 is a decision tree algorithm that extends ID3 by introducing three key mechanisms: attribute selection based on gain, support for continuous attributes, and post-pruning to control overfitting. Recent studies position C4.5 as an entropy-based decision tree algorithm that achieves a balance between classification accuracy and decision interpretability in applied settings [10], [22].

C4.5 constructs a decision tree through recursive data partitioning until sufficient class homogeneity is achieved. Continuous attributes are handled by determining an optimal threshold that maximizes gain, while missing values are addressed through fractional instance weighting across consistent branches. To improve generalization, C4.5 applies post-induction pruning commonly pessimistic error pruning to simplify tree structure without degrading predictive performance [10], [22]. These characteristics make C4.5 suitable for decision-support systems that require transparent and auditable classification rules.

The general stages of the C4.5 algorithm applied in this study are as follows.

1. Preparing a labeled dataset.
2. Selecting the root attribute (the best attribute based on the gain ratio) using the following formula:

$$Entropy(S) = -\sum_{c \in Classes} p_c \log_2 p_c \tag{3}$$

In this formulation, Classes represents the set of all possible classes. The variable $p_{(c)}$ denotes the proportion of objects in the dataset S that belong to class c, where $p_{(c)}$ is calculated as the ratio between $N_{(c)}$ and N. The value $N_{(c)}$ indicates the number of objects in S that are assigned to class c, while N represents the total number of objects contained in the dataset S. The term $\log_2$ refers to the logarithm with base 2 used in the entropy calculation.

For each candidate attribute A, the dataset S is partitioned into several subsets S_v according to the values or intervals of attribute A. The information gain is then calculated as:

$$Gain(S,A) = Entropy(S) - \sum_{v} \frac{|S_v|}{|S|} Entropy(S_v) \quad (4)$$

In this formulation, S denotes the set of data instances at the node currently being processed, while A represents the candidate splitting attribute. The symbol v refers to a specific value or interval of attribute A, and S_v denotes the subset of data instances whose attribute A takes the value v or falls within interval v. The quantity $|S_v|$ indicates the number of instances in subset S_v, whereas $|S|$ represents the total number of instances in the dataset S. The term Entropy(S_v) denotes the entropy of subset S_v.

The attribute with the highest Gain (S,A) value is selected as the splitting attribute at the root of the decision tree.

3. Creating a branch for each value or interval of the selected attribute.
4. Distributing the cases into their corresponding branches.
5. Repeating the process for each branch until the stopping criteria are satisfied.
6. Deriving if–then rules from the resulting decision tree.

2.4 Evaluation Model

Model evaluation was conducted to assess the extent to which the C4.5 algorithm is able to predict laptop feasibility classes derived from the K-Means clustering results. In this stage, the performance of the classification model was evaluated using a confusion matrix and standard evaluation metrics, namely accuracy, precision, recall, and F1-score.

The first step of the evaluation process involved splitting the classification dataset into training data and testing data. The dataset was randomly divided with a proportion of 80% for training and 20% for testing, ensuring that the distribution of feasibility classes was adequately represented in both subsets.

Next, the training data were used to build the C4.5 classification model by learning a decision tree based on the input attributes, including processor type, release year, RAM capacity, storage type, and operating system, with laptop feasibility as the target class. The trained model was then applied to the testing data to generate predicted feasibility labels. At this stage, each testing record contained both the actual class label and the predicted class label produced by the model.

Subsequently, a confusion matrix was constructed by comparing the predicted class labels with the actual class labels from the testing data. The confusion matrix summarizes the number of correct and incorrect predictions for each feasibility class and serves as the basis for computing the evaluation metrics.

Based on the confusion matrix, accuracy, precision, recall, and F1-score were calculated to quantitatively evaluate the performance of the C4.5 model. Accuracy measures the proportion of correctly classified instances and is defined as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (5)$$

Precision represents the proportion of correctly predicted positive instances among all predicted positives:

$$Precision = \frac{TP}{TP + FP} \times 100\% \quad (6)$$

Recall reflects the ability of the model to correctly identify all actual positive instances:

$$Recall = \frac{TP}{TP + FN} \times 100\% \quad (7)$$

The F1-score provides a balanced measure by computing the harmonic mean of precision and recall:

$$F1\text{-}Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \times 100\% \quad (8)$$

In these equations, TP, TN, FP, and FN denote true positives, true negatives, false positives, and false negatives, respectively. Together, these metrics provide a comprehensive assessment of the classification performance and form the basis for the discussion of results in the subsequent section.

3. Results and Discussions

This section presents the results of data processing and analysis conducted to assess laptop feasibility using the C4.5 algorithm, with class labels derived from K-Means clustering. The discussion covers an overview of the dataset, the applied data preprocessing stages, the results of the K-Means clustering and C4.5 classification, and the evaluation of model performance based on the metrics described in the methodology section.

The section begins with a description of the dataset used as the foundation of the study, followed by the data preprocessing procedures, including data cleaning and data transformation. Subsequently, the results of the K-Means algorithm are presented to illustrate the formation of laptop clusters, followed by the classification results obtained using the C4.5 algorithm. Finally, the performance of the proposed model is evaluated to determine its effectiveness in predicting laptop feasibility classes.

3.1 Dataset

The dataset used in this study consists of laptop asset records collected from the IT asset management environment of PT Semen Indonesia. Each record represents a single laptop unit along with its technical specifications, which serve as the basis for feasibility assessment. The dataset includes information related to processing capability, release period, memory capacity, storage technology, and operating system, all of which are commonly considered critical factors in evaluating the usability and lifecycle status of computing devices.

Table 3.1 presents a snapshot of the dataset structure used in this research. The attributes include laptop model identification, processor type, release year, RAM capacity, storage type, and operating system. The dataset comprises a total of 1,905 records, covering a wide range of device generations, from older models released in the late 2000s to recent devices introduced after 2020. This diversity enables the analysis to capture variations in technological maturity and performance characteristics across different laptop categories.

Table 3.1. Dataset Structure for Laptop Feasibility Analysis

No	Model	Processor	Tahun Rilis	RAM (GB)	Storage Type	Operating System
1	Elitebook 830 G5	Intel Core i7	2017	8	SSD	Windows 11
2	S145-4HID	Intel Core i3	2018	4	HDD	Windows 11
3	MacBook Pro M1 2020	Apple M1	2020	8	SSD	MacOS
4	YOGA 300-11IBY	Intel Celeron	2014	2	HDD	Windows 10
5	AO722	AMD C-60	2011	2	HDD	Windows 8.1
…	…	…	…	…	…	…
1901	X453SA	Intel Pentium	2015	4	HDD	Windows 8.1
1902	Elitebook 630 G11	Intel Core Ultra 5	2023	16	SSD	Windows 11
1903	Latitude E5400	Intel Core 2 Duo	2008	2	HDD	Windows 7
1904	Thinkpad X1 Carbon	Intel Core Ultra 7	2017	32	SSD	Windows 11
1905	Macbook Pro M3	Apple M3	2023	16	SSD	MacOS

3.2 Data Preprocessing

Data preprocessing was conducted to prepare the dataset prior to analysis using the K-Means and C4.5 algorithms. This stage aimed to ensure that the data were clean, consistent, and represented in a format compatible with the requirements of clustering and classification methods. Inadequate preprocessing, such as the presence of missing values, inconsistent attribute formatting, or duplicate records, may degrade model performance and complicate result interpretation.

In this study, data preprocessing consisted of two main steps: data cleaning and data transformation. These steps serve as an essential bridge between the raw dataset and the subsequent analytical stages involving K-Means clustering and C4.5 classification.

1. Data cleaning
   Data cleaning was performed to remove irrelevant, incomplete, duplicate, and inconsistent records. From the initial dataset containing 1,905 laptop records, only five attributes relevant to feasibility assessment were retained, namely processor type, release year, RAM capacity, storage type, and operating system, while the model identifier attribute was excluded. Text-based attributes were standardized to eliminate formatting inconsistencies, and duplicate records were removed so that each remaining entry represented a unique combination of laptop specifications. As a result of this process, the dataset was reduced to 85 unique records, reflecting distinct laptop specification profiles free from duplication and inconsistency.
2. Data Transformation
   Data transformation was subsequently applied to convert all attributes into numerical representations with comparable scales. Categorical attributes, including processor type, storage type, and operating system, were encoded using an ordinal performance-based scoring scheme and then normalized to the range [0,1] using Min–Max scaling. This approach preserves the relative performance hierarchy among categories while enabling numerical processing. Numerical attributes, namely release year and RAM capacity, were normalized using Min–Max scaling to ensure proportional contribution across features. Through this transformation, newer laptops and devices with higher memory capacity are represented by higher normalized values, while older or lower-specification devices receive lower values.

After preprocessing, all attributes were represented in a consistent numerical scale, making the dataset suitable for distance-based clustering and decision tree classification. The final preprocessed dataset used in this study is summarized in Table 3.2.

Table 3.2 Data Preprocessing

No	Processor	Tahun Rilis	RAM (GB)	Storage Type	Operating System
1	0.75	0.588235	0.2	1	1
2	0.25	0.647059	0.066667	0	1
3	0.75	0.764706	0.2	1	1
4	0	0.411765	0	0	0.5
5	0	0.235294	0	0	0
…	…	…	…	…	…
1901	0	0.470588	0.066667	0	0
1902	0.5	0.941176	0.466667	1	1
1903	0	0.058824	0	0	0
1904	0.75	0.588235	1	1	1
1905	1	0.941176	0.466667	1	1

3.3 K-Means Algorithm

The K-Means algorithm was applied to group laptops based on the similarity of their technical specifications. The analysis was conducted on the preprocessed dataset consisting of 85 records and five numerical feature attributes, namely processor performance score, release year, RAM capacity, storage type, and operating system edition. All attributes were normalized to a comparable scale prior to clustering to ensure valid distance-based computation.

In this study, the number of clusters was fixed at K=3, reflecting the conceptual categorization of laptop feasibility into three operational groups: feasible for use, requires upgrade, and requires replacement. This choice was driven by practical asset management considerations rather than optimization-based cluster number selection.

The clustering process was initialized using three representative centroids corresponding to low, medium, and high specification profiles. These initial centroids were selected from existing data points to reflect extreme and intermediate combinations of laptop specifications, thereby providing a meaningful starting point for the iterative clustering process. Euclidean distance was used to measure similarity between each data object and the centroids.

During each iteration, all data points were assigned to the nearest centroid based on Euclidean distance, after which new centroids were computed as the mean of the feature vectors within each cluster. This assignment–update process was repeated iteratively until convergence was achieved. In this study, convergence occurred at the fourth iteration, as indicated by stable cluster membership across successive iterations. The final cluster composition consisted of 16 records in cluster 0, 47 records in cluster 1, and 22 records in cluster 2.

Following convergence, the clusters were analyzed to identify their characteristic patterns and to assign meaningful operational labels. The largest cluster was dominated by laptops with relatively recent release years, moderate to high RAM capacity, solid-state storage, and updated operating systems, indicating devices that remain suitable for continued use. This cluster was labeled Feasible for Use. Another cluster exhibited intermediate characteristics, including moderate release years, mixed storage types, and sufficient but not optimal memory capacity, suggesting laptops that remain usable but would benefit from hardware upgrades; this cluster was labeled Requires Upgrade. The remaining cluster consisted primarily of older laptops with lower RAM capacity, hard disk storage, and outdated operating systems, representing devices that are no longer technically adequate and therefore labeled Requires Replacement.

The complete clustering results, including the final cluster assignments and feasibility labels for all laptop specification profiles, are summarized in Table 3.3. These cluster labels serve as the basis for the subsequent classification stage using the C4.5 algorithm.

3.4 C4.5 Algorithm

After the K-Means clustering process produced three laptop feasibility groups, the next stage of the analysis was to construct a classification model using the C4.5 algorithm. In this stage, the cluster labels obtained from K-Means were treated as class labels, enabling the C4.5 algorithm to learn decision patterns that explain why a laptop belongs to a particular feasibility category based on its technical specifications. The primary objective of this stage was to generate interpretable decision rules that can support transparent and consistent feasibility assessment.

The classification dataset consisted of 85 records obtained from the clustering results. Each record represented a unique combination of laptop specifications. To facilitate entropy and information gain calculations while preserving technical interpretability, numerical attributes were mapped back into ordinal categorical levels based on performance scores. The target variable comprised three feasibility classes, namely Feasible for Use, Requires Upgrade, and Requires Replacement.

At the root node of the decision tree, entropy and information gain were computed for all candidate attributes, including processor, release year, RAM capacity, storage type, and operating system. The results show that storage type yielded the highest information gain, indicating that this attribute provides the greatest reduction in class uncertainty. Consequently, storage type was selected as the root attribute of the decision tree. All laptops equipped with solid-state drives (SSD) were consistently classified into the Feasible for Use category, resulting in a homogeneous branch that required no further splitting.

In contrast, the branch corresponding to hard disk drive (HDD) storage remained heterogeneous, containing laptops classified as either Requires Upgrade or Requires Replacement. Further analysis of this subset revealed that operating system was the most informative attribute for additional partitioning. Splitting the HDD branch based on operating system categories produced fully homogeneous branches: laptops with medium- or high-level operating systems were classified as Requires Upgrade, while laptops with low-level operating systems were classified as Requires Replacement. Since all branches achieved class homogeneity at this stage, the recursive splitting process terminated.

The final C4.5 decision tree is presented in Figure 4.2.

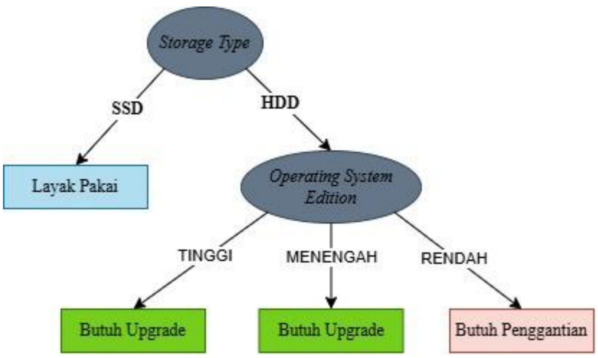

Figure 3.1 C4.5 Decision tree

Figure 4.2 visually summarizes the classification logic derived from the dataset. The tree structure highlights storage type as the primary decision factor, while operating system functions as a secondary discriminator for laptops that use HDD storage. This structure demonstrates that a small number of technically meaningful attributes is sufficient to explain laptop feasibility in a clear and interpretable manner.

Based on the final structure of the decision tree, the resulting if–then decision rules can be expressed as follows:

1. If the storage type of a laptop is SSD, then the laptop is classified as Feasible for Use.
2. If the storage type of a laptop is HDD and the operating system belongs to a medium or high category, then the laptop is classified as Requires Upgrade.
3. If the storage type of a laptop is HDD and the operating system belongs to a low category, then the laptop is classified as Requires Replacement.

These numbered if–then rules represent the final output of the C4.5 classification process. The rules are concise, transparent, and easy to audit, making them suitable for direct implementation as decision guidelines or standard operating procedures to support laptop asset management at PT Semen Indonesia.

3.5 Evaluation Model

Model evaluation was conducted to assess the performance of the C4.5 classification model in predicting laptop feasibility classes derived from the K-Means clustering results. The evaluation was performed using testing data that were not involved in the training process to ensure an objective assessment of the model's predictive capability.

The dataset was divided into training and testing subsets using an 80:20 split while maintaining class distribution. The trained C4.5 model was then applied to the testing data to generate predicted feasibility labels. Model performance was evaluated using a confusion matrix and standard classification metrics, including accuracy, precision, recall, and F1-score.

The evaluation results show that the C4.5 model achieved an overall accuracy of 100% on the testing data. All testing instances were correctly classified into their respective feasibility classes, namely Requires Replacement, Requires Upgrade, and Feasible for Use. Precision, recall, and F1-score for each class also reached 1.00, indicating perfect classification performance across all categories.

The confusion matrix illustrating the classification results is presented in Figure 3.2.

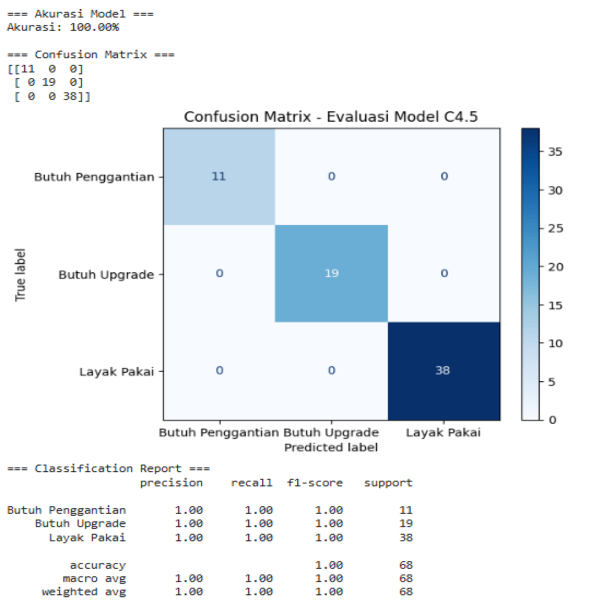

Figure 3.2 Evaluation Model

3.3 Limitations and Future Work

Although the proposed framework demonstrates excellent performance in classifying laptop feasibility, several limitations of this study should be acknowledged. First, the dataset was obtained from a single organizational environment, which may limit the generalizability of the results to other institutions with different asset characteristics, usage patterns, or technology standards. Second, the number of clusters in the K-Means stage was determined based on practical feasibility categories rather than optimized through cluster validity indices, which may overlook more granular patterns within the data.

Another limitation relates to the transformation of categorical attributes into ordinal performance scores. While this approach preserves technical interpretability, it relies on expert judgment and may introduce subjectivity into the clustering and classification processes. In addition, the evaluation results showing perfect classification performance were obtained on a relatively limited dataset, which may not fully reflect the variability encountered in larger or more heterogeneous asset inventories.

Future research can address these limitations by incorporating datasets from multiple organizations and expanding the number of laptop records to improve robustness and external validity. Data-driven techniques for determining the optimal number of clusters, as well as alternative encoding strategies for categorical attributes, may also be explored. Furthermore, integrating additional variables such as device usage intensity, maintenance history, and performance degradation over time could enhance the practicality of the feasibility assessment. Future studies may also compare the proposed approach with ensemble or hybrid classification models to balance predictive accuracy and interpretability in broader IT asset management contexts.

4. Conclusions

This study presented an integrated framework for assessing laptop feasibility by combining K-Means clustering and C4.5 decision tree classification based on technical specifications commonly available in IT asset inventories. The results demonstrate that the proposed approach is able to systematically group laptops into three feasibility categories (Feasible for Use, Requires Upgrade, and Requires Replacement) and subsequently derive interpretable decision rules that explain these categories in a transparent manner. The K-Means algorithm successfully captured meaningful patterns in the preprocessed dataset, showing clear differentiation among laptops with high, medium, and low technical capabilities. These clustering results directly address the research objective of identifying objective feasibility groupings without relying on subjective judgment. Building upon these groupings, the C4.5 algorithm effectively learned classification rules that map laptop specifications to feasibility classes, with storage type and operating system emerging as the most influential attributes in the decision structure. The evaluation results further confirm the effectiveness of the proposed framework, as the classification model achieved perfect performance on the testing dataset, indicating strong consistency between the clustering outcomes and the derived decision rules. The primary contribution of this research lies in the integration of unsupervised clustering and supervised classification to produce both data-driven feasibility labels and explainable rules that can be easily audited and implemented. Unlike approaches that focus solely on prediction accuracy, this study emphasizes interpretability, making the results suitable for practical decision-making in IT asset management. From a practical perspective, the generated if–then rules can be directly translated into operational guidelines or standard operating procedures to support consistent and transparent decisions regarding laptop usage, upgrades, and replacements. From a broader perspective, this work illustrates how combining clustering and decision tree techniques can enhance explainability in asset evaluation tasks, contributing to the application of data mining methods in organizational decision-support systems. Future research directions include validating the framework on larger and more diverse datasets, incorporating additional operational factors such as usage intensity and maintenance history, and exploring hybrid or ensemble models to further balance interpretability and predictive robustness.

References

[1]. M. Jasiulewicz-Kaczmarek, A. Saniuk, and T. Nowicki, “Asset lifecycle management and sustainability-oriented decision-making in industrial systems,” Sustainability, vol. 15, no. 4, pp. 1–18, 2023.

[2]. B. Bhurtel and D. Rawat, “IT asset lifecycle management and risk-based decision support,” Journal of Information Systems Management, vol. 40, no. 2, pp. 123–137, 2023.

[3]. J. Kim, H. Lee, and S. Park, “Data-driven asset management using machine learning techniques,” IEEE Access, vol. 12, pp. 45678–45690, 2024.

[4]. C. Parra, L. Martínez, and J. Ruiz, “Lifecycle-based decision models for information technology assets,” Decision Support Systems, vol. 176, pp. 113915, 2024.

[5]. A. Darmawan, R. Pratama, and M. Hidayat, “Application of data mining for IT asset feasibility analysis,” Journal of Information Technology Systems, vol. 9, no. 1, pp. 45–56, 2024.

[6]. A. Rindu, S. Wahyuni, and F. Nugroho, “Decision support systems for hardware replacement planning,” Indonesian Journal of Computing, vol. 8, no. 2, pp. 89–101, 2023.

[7]. T. Bold and G. Urschel, “Managing IT assets through analytics and predictive models,” Information Systems Management, vol. 40, no. 3, pp. 201–214, 2023.

[8]. J. Ortega-Guzmán, M. López, and A. Torres, "Knowledge discovery frameworks for asset evaluation," Expert Systems with Applications, vol. 237, pp. 121393, 2024.

[9]. A. H. Pitafi, M. Kanwal, and A. Khan, "Performance analysis of K-Means clustering in applied data mining," Applied Soft Computing, vol. 136, pp. 110087, 2023.

[10]. H. Blockeel, J. Vanschoren, and A. Bifet, "Decision tree learning: Recent advances and practical considerations," Machine Learning, vol. 112, no. 7, pp. 2761–2791, 2023.

[11]. M. Jusia, R. Siregar, and D. Putra, "Optimization of C4.5 decision tree using metaheuristic approaches," Journal of Computer Science and Applications, vol. 15, no. 3, pp. 210–221, 2024.

[12]. M. Golazad, A. Rahimi, and H. Karimi, "A comprehensive review of data preprocessing techniques for machine learning," Data Science and Analytics, vol. 9, no. 1, pp. 1–20, 2024.

[13]. M. Ibrahimi, A. Bouali, and L. Boussaid, "Feature scaling and encoding strategies in machine learning models," Journal of Big Data, vol. 10, no. 1, pp. 1–22, 2023.

[14]. X. Gong, Y. Li, and Z. Wang, "Impact of data quality and preprocessing on machine learning performance," Knowledge-Based Systems, vol. 268, pp. 110407, 2023.

[15]. A. Pinheiro, R. Silva, and P. Costa, "Evaluation of feature normalization techniques in classification tasks," Pattern Recognition Letters, vol. 176, pp. 1–9, 2025.

[16]. M. Côté, J. M. Bouchard, and F. Saubion, "A systematic survey on data cleaning for machine learning," ACM Computing Surveys, vol. 56, no. 2, pp. 1–36, 2024.

[17]. A. Borrohou, K. El Yassini, and M. Azizi, "Outlier detection and handling in data preprocessing," Procedia Computer Science, vol. 219, pp. 328–335, 2023.

[18]. N. Wongoutong, S. Thongkam, and P. Meesad, "Effect of feature scaling on distance-based clustering algorithms," International Journal of Data Mining & Knowledge Management Process, vol. 14, no. 1, pp. 15–27, 2024.

[19]. K. Rashmi, R. Kumar, and S. Patel, "Comparative study of normalization techniques for clustering algorithms," Journal of Intelligent Systems, vol. 33, no. 1, pp. 245–258, 2024.

[20]. J. Han, M. Kamber, and J. Pei, Data Mining: Concepts and Techniques, 3rd ed. San Francisco, CA, USA: Morgan Kaufmann, 2012.

[21]. S. Lee, J. Park, and H. Choi, "Z-score normalization effects on K-Means clustering for time-series data," IEEE Access, vol. 12, pp. 99871–99883, 2024.

[22]. D. Ulqinaku and S. Ktona, "Stability and interpretability of gain ratio–based decision trees," Artificial Intelligence Review, vol. 57, no. 1, pp. 1–29, 2024.

Biographies of Authors

Fachriqi Naldes was born on January 22, 1995, in Padang, Indonesia. He received his bachelor's degree in Mechanical Engineering from Universitas Andalas in 2019. He is currently employed at PT Sinergi Informatika Semen Indonesia and is stationed at PT Semen Padang, located on Raya Indarung Street, Indarung Subdistrict, Lubuk Kilangan District, Padang City, West Sumatra, Indonesia. His professional experience focuses on information technology asset management, particularly in the areas of asset lifecycle management. He can be contacted at email: fachriqinaldes@gmail.com.

Sumijan was born in Nganjuk on May 7 1966. He received the Bachelor Degree in Informatics Management in 1991 from Universitas Putra Indonesia YPTK, Master of Information Technology in 1998 from University Technology Malaysia (UTM). He completed has Doctorate of Information Technology as Medical Image Expertise from Gunadarma University in December 2015. He is member of ACM (23145751). He can be contacted at e-mail: sumijan@upiyptk.ac.id.

Syafri Arlis was born in Padang on October 23, 1986. His undergraduate study was completed in 2009 at Universitas Putra Indonesia YPTK. He completed his Masters degree at Putra Indonesia University, YPTK Padang. He currently serves as a lecture in the Informatics Engineering study program at the Universitas Putra Indonesia YPTK Padang. Teaching history that has been carried out starting from 2011 until now, such as database and digital image processing. Published research history places more emphasis on the atificial neuralnetwork and digital image processing. He can be contacted at email: syafri_arlis@upiyptk.ac.id.