
Application Of The K-Means Clustering Method In Grouping Goat Productivity In A Web-Based Monitoring System

Hafizh Trisna Indarto¹⁾, Ismail Abdurrozzaq Zulkarnain²⁾, Arin Yuli Astuti³⁾
^{1,2,3)} Muhammadiyah University of Ponorogo

*Corresponding Author
Email: trishnahafizh@gmail.com

Abstract

Monitoring goat productivity at Pawit Mulya Farm Pacitan was previously hampered by manual data management and the lack of systematic productivity grouping. This study aims to develop a web-based monitoring system by implementing K-Means Clustering based on body weight, birth rate, and milk production. The study used a system development approach with the Waterfall method. The study population consisted of approximately 230 goats, while the sample consisted of productivity data that met the research variables. Data collection was carried out through observation, interviews, and literature studies. Analysis used Euclidean Distance, centroid updates, convergence testing, and Silhouette Score. The results of the study produced three productivity clusters: low, medium, and high, with a Silhouette Score of 0.3312, indicating sufficient clustering quality. White Box testing produced five valid independent paths with a 100% success rate. The system can be used as a means of monitoring and grouping productivity in a structured manner, although separation between clusters still needs to be optimized.

Keywords: K-Means Clustering, Livestock Monitoring, Silhouette Score, Web-Based System, Goat Productivity

INTRODUCTION

Goat productivity is the ability of livestock to produce output in the form of body weight gain, birth rate, and milk production. Productivity is influenced by genetics, physiology, feed, maintenance, health, environment, and farm management. One goat breed widely cultivated in Indonesia is the Etawa Crossbred (PE) goat due to its good production and reproductive capabilities, adaptability to tropical climates, and high economic value (Syadik & Salawati, 2021; Pebriadi & Amalyadi, 2025).

Pawit Mulya Farm Pacitan is a PE goat farm with around 230 goats. Monthly livestock performance records are conducted, including body weight, birth rate, and milk production. The availability of this data is crucial for livestock management, particularly to support the monitoring and evaluation of goat productivity (Rosyidah et al., 2022).

However, the recording process at Pawit Mulya Farm is still manual, making it prone to errors, inefficient, and complicating data retrieval and evaluation. This situation prevents productivity data from being optimally utilized to support livestock management and decision-making (Syadik & Salawati, 2021; Nugroho, 2024; Ikhsan & Purnomo, 2023).

Another problem is the lack of classification of goat productivity into low, medium, and high categories. This makes it difficult for farmers to identify the productivity of each goat as a basis for evaluation and husbandry strategies. Therefore, more structured and integrated data management is needed.

This study aims to develop a web-based goat productivity monitoring system by applying the K-Means Clustering algorithm based on body weight, birth rate, and milk production. K-Means was chosen because it has a simple and efficient calculation process for grouping numerical data based on similarity using Euclidean distance (Ndasak et al., 2023; Rilvani, 2025). The urgency of this research lies in the need for digitizing livestock productivity recording and monitoring, while its novelty lies in the application of K-Means Clustering in a web-based monitoring system at Pawit Mulya Farm Pacitan to produce low, medium, and high productivity groups.

RESEARCH METHODS

This research is a system development study with an implementative approach to build a web-based goat productivity monitoring system at Pawit Mulya Farm, Pacitan. The system development method used is the Waterfall approach, consisting of problem identification, data collection, needs analysis, design, implementation, testing, and maintenance. The system applies the K-Means Clustering algorithm to group goat productivity into low, medium, and high categories (Ndasak et al., 2023; Nugroho, 2024).

The study population consisted of all 230 goats raised at Pawit Mulya Farm in Pacitan. The sample used consisted of goat productivity data that met the research variables, namely gender, body weight, birth rate, and milk production. The manual calculation example used data from 20 goats, coded K001 to K020, to demonstrate the K-Means calculation process.

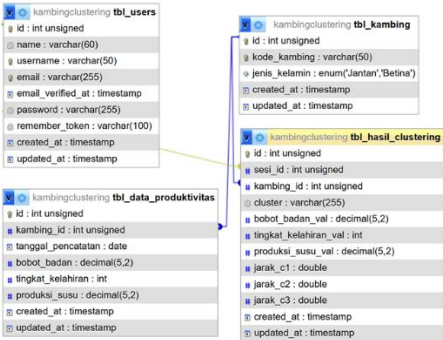
The research instruments consisted of hardware and software for system development, while data collection was conducted through observation, interviews, and literature review. Observations were conducted at Pawit Mulya Farm, Pacitan, to determine the condition and process of monitoring goat productivity. Interviews were conducted with Mr. Boniran, the farm owner, to obtain information on livestock management, productivity data, problems, and system requirements. The literature review was conducted through journals, books, articles, and previous research related to goat productivity, livestock information systems, and K-Means Clustering (Syadik & Salawati, 2021; Rosyidah et al., 2022; Rilvani, 2025).

Data analysis was performed using the K-Means Clustering algorithm with variables of sex, body weight, birth rate, and milk production. Sex was converted into numeric data, namely 0 for males and 1 for females, then the data was grouped into three clusters: low, medium, and high. The process was carried out by determining the initial centroid, calculating the distance using Euclidean Distance, determining the closest cluster, calculating the new centroid, and repeating the process until the clusters were stable (Ndasak et al., 2023; Rilvani, 2025). System testing used White Box Testing with Cyclomatic Complexity to identify independent paths that needed to be tested.

The research procedure uses the Waterfall method, starting from problem identification, data collection, needs analysis, system design, implementation, testing, and maintenance. The system design includes flowcharts, Use Case Diagrams, DFDs, ERDs, databases, and interfaces. In the implementation phase, a web-based system was built and the K-Means algorithm was applied to group goat productivity. Next, the system was tested using White Box Testing, followed by maintenance in the form of error corrections, feature updates, database checks, and adjustments to user needs.

RESULTS AND DISCUSSION

Database Implementation



Picture 1.Database Table Relations

Implementation of the K-Means Algorithm

```
$kambings = kambing::with('produktivitas')->get();

foreach ($kambings as $kambing) {
    $avgBobot = $kambing->produktivitas-
>avg('bobot_badan') ?? 0;
    $maxKelahiran = $kambing->produktivitas-
>max('tingkat_kelahiran') ?? 0;
    $avgSusu = $kambing->produktivitas-
>avg('produksi_susu') ?? 0;

    $dataset[$kambing->id] = [
        'C1' => (float)$avgBobot,
        'C2' => (float)$maxKelahiran,
        'C3' => (float)$avgSusu ];}
```

Figure 1. Dataset Retrieval

Based on the source code of Figure 1, goat data is processed into body weight values, birth rates, and milk production which are stored as datasets C1, C2, and C3 for the K-Means Clustering process.

```
$centroids = [
    'Rendah' => $kambingRendah['scores'],
    'Sedang' => $kambingSedang['scores'],
    'Tinggi' => $kambingTinggi['scores']
];
$history = [];
$converged = false;
$iteration = 0;
```

Figure 2. Determining the Initial Centroid

Based on the source code of Figure 2, the system determines three initial centroids for the Low, Medium, and High clusters based on the percentile method, then initializes the history, convergence, and number of iterations for the K-Means process.

```
private function calculateEuclideanDistance(array
$point1, array $point2): float
{
    $sum = pow($point1['C1'] - $point2['C1'], 2) +
    pow($point1['C2'] - $point2['C2'], 2)+
    pow($point1['C3'] - $point2['C3'], 2);

    return sqrt($sum);
}
```

Figure 3. Euclidean Distance

Based on the source code in Figure 3, the calculateEuclideanDistance() function calculates the distance between goat data and the centroid based on body weight, birth rate, and milk production. The distance value is calculated using the Euclidean Distance formula, then the data is assigned to the cluster with the smallest distance.

```
$distC1 = $this-
>calculateEuclideanDistance($scores,
$centroids['Rendah']);
$distC2 = $this-
>calculateEuclideanDistance($scores,
$centroids['Sedang']);
$distC3 = $this-
>calculateEuclideanDistance($scores,
$centroids['Tinggi']);

$minDist = min($distC1, $distC2, $distC3);

if ($minDist === $distC1) {
    $cluster = 'Rendah';
} elseif ($minDist === $distC2) {
    $cluster = 'Sedang';
} else {
    $cluster = 'Tinggi';
}
```

Figure 4. Cluster Determination

Based on the source code in Figure 4, the system calculates the distance of the data to three centroids: Low, Medium, and High. The data is then assigned to the cluster with the smallest distance value through an if...elseif...else branch.

```
$newCentroids[$clusterName] = [
    'C1' => round($sumC1 / $count, 4),
    'C2' => round($sumC2 / $count, 4),
    'C3' => round($sumC3 / $count, 4)
];
```

Figure 5. Cluster Update

Based on the source code in Figure 5, the system calculates a new centroid based on the average of attributes C1, C2, and C3 in each cluster. The result is rounded and used as the centroid in the next iteration until it remains unchanged.

```
$converged = $this->checkCentroidsConvergence($centroids, $newCentroids);

$history[] = $iterationLog;

$centroids = $newCentroids;
```

Figure 6. Convergence Function

Based on the source code in Figure 6, the system calls the check Centroids Convergence function to compare the old centroid value with the latest calculated centroid value.

```
private function
checkCentroidsConvergence(array
    $oldCentroids, array $newCentroids): bool
{
    foreach ($oldCentroids as $cluster =>
        $coords) {
        if (
            $coords['C1'] !=
            $newCentroids[$cluster]['C1'] ||
            $coords['C2'] !=
            $newCentroids[$cluster]['C2'] ||
            $coords['C3'] !=
            $newCentroids[$cluster]['C3']
        ) {
            return false;
        }
    }
    return true;
}
```

Figure 7. Convergence Check

Based on the source code in Figure 7, the checkCentroidsConvergence() function compares the old and new centroids on attributes C1, C2, and C3. If the centroids change, the iteration continues; if they do not, the clustering process is declared converged.

```
foreach ($dataset as $kambingId => $scores) {
    $detail =
    $finalIteration['details'][$kambingId];

    HasilClustering::create([
        'sesi_id' => $sesi->id,
        'kambing_id' => $kambingId,
        'cluster' => $detail['kluster_sementara'],
        'bobot_badan_val' => $scores['C1'],
        'tingkat_kelahiran_val' =>
        (int)$scores['C2'],
        'produksi_susu_val' => $scores['C3'],
        'jarak_c1' => $detail['jarak_c1'],
        'jarak_c2' => $detail['jarak_c2'],
        'jarak_c3' => $detail['jarak_c3'],
    ]);
}
```

Figure 8. Clustering Results

Based on the source code in Figure 8, the system saves the clustering results for each goat to the hasil_clustering table, including the goat ID, cluster, productivity value, and distance to the centroid. This data is used to display clustering results and reports.

```
print("\n*40")
print("HASIL EVALUASI CLUSTERING")
print("\n*40")
print(f"Jumlah Cluster : 3")
print(f"Silhouette Score : {score:.4f}")

=====
HASIL EVALUASI CLUSTERING
=====
Jumlah Cluster : 3
Silhouette Score : 0.3312
```

Figure 9. Silhouette Score Results

Based on Figure 9, the Silhouette Score evaluation yielded a value of 0.3312, indicating sufficient clustering quality. These results can be used as a basis for monitoring

and evaluating goat productivity.
System Implementation Results

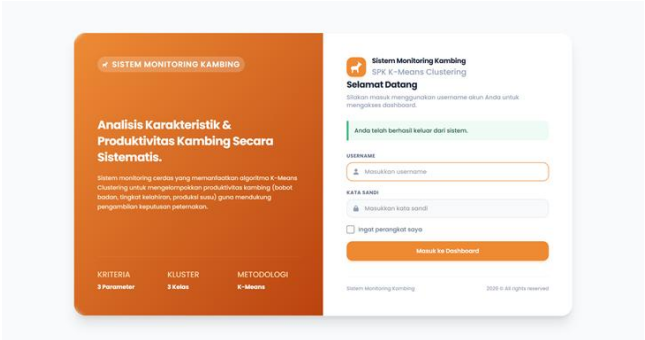


Figure 10. Login page

Figure 10 shows the login page for initial system access. Users enter their username and password for authentication before accessing the dashboard. This page also displays brief information about the application of K-Means Clustering in grouping goat productivity.

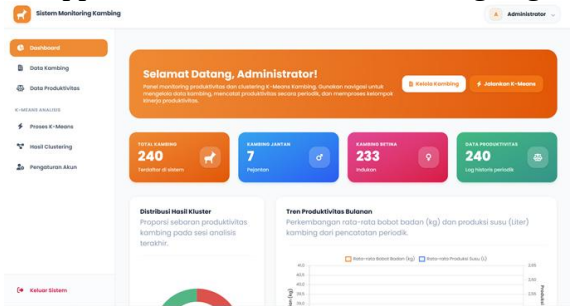


Figure 11. Dashboard page

Figure 12 shows a dashboard page displaying a summary of data from 240 goats, 253 productivity data, and a visualization of the K-Means results. Data management, clustering, and account menus are available, allowing users to quickly and systematically monitor goat productivity.

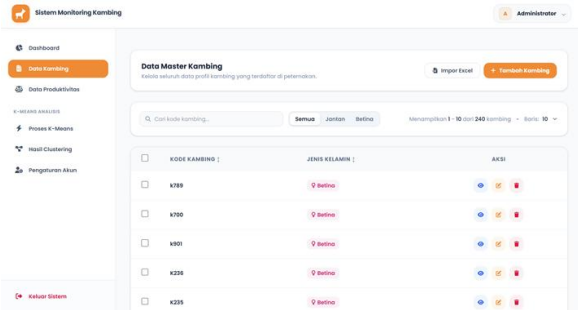


Figure 12. Goat data page

Figure 13 shows the Goat Data page used to manage goat identities. Users can search, filter, add, import, modify, delete, and view goat data details based on code and gender.

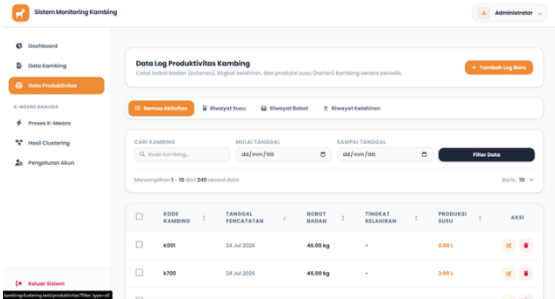


Figure 13. Productivity data page

Figure 14 shows the Productivity Data page used to manage body weight, birth rate, and milk production data. Users can add, search, edit, delete, and filter data by goat and time period. This data serves as input for the K-Means Clustering process.

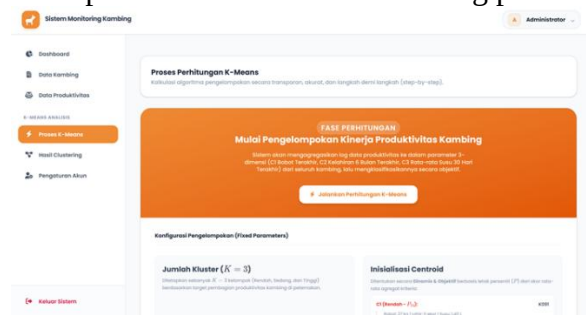


Figure 14. K-Means Clustering Process

Figure 14 shows the K-Means Clustering process page used to group goat productivity based on body weight, birth rate, and milk production. The system uses three clusters: low (C1), medium (C2), and high (C3), then calculates the Euclidean distance and iterates until the centroids are stable.

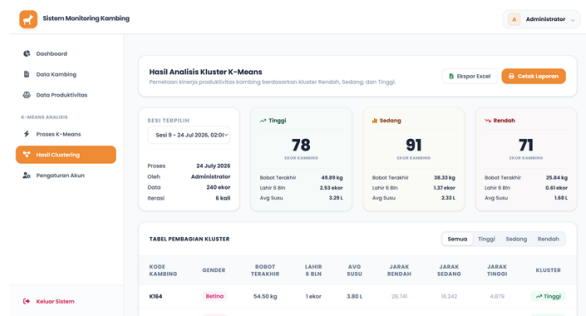


Figure 15. Clustering results page

Based on figure 15, the goat with code K164 is included in the tall category.

Table 1.Final Centroid Table

Cluster	Weight	Birth	Milk
C1 Low	37.29	1	2.06
C2 Medium	44.33	2	2.43
C3 High	53.54	2	2.49

In Table 1, the centroid is the center point of each cluster, obtained after the K-Means iteration process is complete. This centroid value is used as a reference for calculating the level of closeness of each goat data to each cluster.

Table 2. Cluster Determination Results Table

Cluster	Distance
Low	17.87
Currently	10.88
Tall	4.76

In Table 2, the smallest distance is in the High Cluster (C3) at 4.76, so goat K164 is in the High Cluster. This result is consistent with the system and indicates that the K-Means implementation has run correctly.

White Box system testing

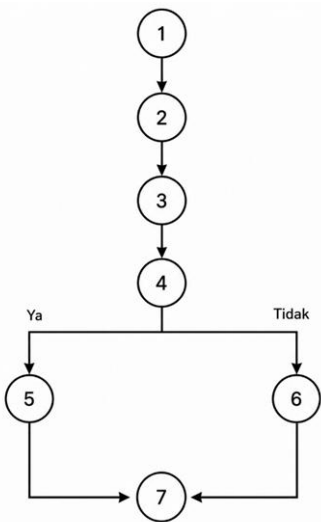


Figure 16. White Box Login Module Test Results

Figure 16 shows a flowgraph of the Login module with 7 nodes and 8 edges. The Cyclomatic Complexity calculation of 2 yields two independent paths, both of which are successfully tested and valid. This indicates that the Login module is running according to the program's logic.

Table 3. Login Module Test Results

Test Track	Expected results	Actual Results	Status
Path 1	The system displays the Dashboard	In accordance	Valid
Path 2	The system displays a login failed message.	In accordance	Valid

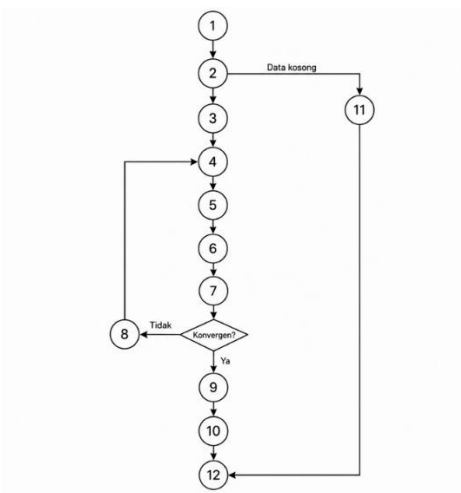


Figure 17. White Box Test Results of the K-Means Clustering Module

Figure 17 shows the flowgraph of the K-Means Clustering module with a Cyclomatic Complexity value of 3, resulting in three independent paths being tested. All paths were successfully executed and valid, indicating that the clustering process has run according to the system design.

Table 4. Clustering Module Test Results

Test Track	Expected results	Actual Results	Status
Path 1	The system generates productivity clusters	In accordance	Valid
Path 2	The system displays an empty data message.	In accordance	Valid
Path 3	The system completed the clustering process without error.	In accordance	Valid

Table 2. Recapitulation of White Box Results

Module	Cyclomatic Complexity	Number of Paths	Status
Login	2	2	Valid
K-Means Clustering	3	3	Valid
Total	5	5	Valid

In Table 4, all independent paths in the login and K-Means Clustering modules were successfully tested and declared valid. These results indicate that the program's logic and branching work as designed, allowing the system to be used for monitoring and clustering goat productivity.

System Recapitulation Results

The test success percentage is used to measure the system's success rate based on the results of tests performed on each feature. The calculation is performed by comparing the number of successful test scenarios to the total number of test scenarios, then expressed as a percentage. This percentage provides an overview of the system's level of functionality in meeting the designed requirements. The success percentage is calculated using the following formula.

$$\text{Persentase Keberhasilan} = \left(\frac{\text{Jumlah Pengujian Berhasil}}{\text{Jumlah Seluruh pengujian}} \right) \times 100\%$$

Based on the recapitulation results of the White Box testing in Table 4.13, 5 test paths were obtained consisting of 2 paths in the Login module and 3 paths in the K-Means Clustering module. All paths were successfully executed according to the program logic flow and obtained valid status, so the number of successful paths is 5 out of a total of 5 paths tested. Based on the calculation results, the percentage of test success can be calculated as follows.

$$\begin{aligned}
 &= \left(\frac{5}{5} \right) \times 100\% \\
 &= 100\%
 \end{aligned}$$

Based on the calculation results, the test success rate was 100%. This value indicates that all 5 tested paths, consisting of 2 paths in the Login module and 3 paths in the K-Means Clustering module, were successfully executed according to the designed program logic flow. All paths obtained valid status, so no errors were found in the program logic flow during the White Box testing process. Thus, the system implementation has met functional requirements and can run according to the established design.

Thus, the test results indicate that the application of the K-Means Clustering algorithm to web-based goat productivity monitoring has met the system's functional requirements. All independent paths have been successfully tested according to the designed program logic, thus the system is declared suitable for use in supporting the monitoring and clustering of goat productivity at Pawit Mulya Farm, Pacitan.

CONCLUSION

This study successfully applied the K-Means Clustering algorithm to a web-based goat productivity monitoring system at Pawit Mulya Farm Pacitan using body weight, birth rate, and milk production variables. The clustering process resulted in three productivity categories: low, medium, and high, with a Silhouette Score of 0.3312 indicating sufficient clustering quality. White Box testing yielded a Cyclomatic Complexity of 2 for the Login module and 3 for the K-Means Clustering module, with all 5 independent paths successfully tested and achieving a valid status or 100% success rate. Practically, the system can help farmers record, manage, monitor, and group goat productivity in a more structured and objective manner, thereby supporting evaluation and decision-making in livestock management.

This study still has limitations, particularly the use of productivity variables limited to body weight, birth rate, and milk production, as well as the Silhouette Score value, which indicates that the separation between clusters is not optimal. Furthermore, the study has not compared K-Means with other clustering methods and has not tested the system over a longer period of use. Future research is recommended to add variables such as feed consumption, health, age, genetics, and reproductive history, as well as compare several clustering algorithms to obtain a method with better clustering quality. System development can also be directed at integrating productivity prediction features, notifications, and periodic data-based analysis so that the system functions not only as a monitoring medium, but also as a more comprehensive livestock decision-support tool.

REFERENCES

- Ikhsan, A. N., & Purnomo, P. (2023). Pemanfaatan smart farming dan digitalisasi untuk peternakan dan produk turunannya. *SWAGATI: Journal of Community Service*, 1(2), 91–96. <https://doi.org/10.24076/swagati.2023v1i2.1142>
- Ndasak, B. O., Batarius, P., & Siki, Y. C. H. (2023). Klasterisasi data hasil produksi pertanian dan peternakan Provinsi Nusa Tenggara Timur menggunakan metode K-Means. *Jurnal Informatika Polinema*, 9(4), 415–426. <https://doi.org/10.33795/jip.v9i4.1369>
- Nugroho, D. P., & Sutopo, J. (2024). Analisis dan pengembangan sistem pengelolaan peternakan kambing berbasis web dengan metode PIECES. *REMik: Riset dan E-Jurnal Manajemen Informatika Komputer*, 8(1), 82–95. <https://doi.org/10.33395/remik.v8i1.13248>
- Pebriadi, W., Hermansyah, & Amalyadi, R. (2025). Analisis pendapatan usaha kambing Peranakan Etawa (PE) di Desa Boyemare Kecamatan Sakra Barat Kabupaten Lombok Timur. *Integrated and Sustainable Animal Production Innovation*, 2(1), 41–54. <https://doi.org/10.29303/i-sapi.v2i1.6837>
- Rilvani, E. (2025). Studi literatur: Perbandingan algoritma K-Means dan Fuzzy C-Means dalam analisis clustering. *Journal of Computer Science and Technology (JCS-TECH)*, 5(2), 52–58. <https://doi.org/10.54840/jcstech.v5i2.382>
- Rosyidah, F., Nugraha, F., & Handayani, P. K. (2022). Sistem informasi pengelolaan data ternak (skala usaha mikro) di Dinas Pertanian dan Pangan Kabupaten Kudus berbasis web. *Journal of Computer and Information Systems Ampera*, 3(3), 176–192. <https://doi.org/10.51519/journalcisa.v3i3.289>
- Syadik, F., Nurmala, & Salawati. (2021). Studi produktivitas kambing pada peternakan rakyat di Kabupaten Tolitoli. *Jurnal Peternakan Nusantara*, 7(2), 79–88. <https://doi.org/10.30997/jpn.v7i2.4572>