INTERNATIONAL JOURNAL OF APPLIED ARTIFICIAL INTELLIGENCE AND ROBOTICS
E-ISSN: 3124-1212, DOI: 00.000/XXXX.YYYY.II.NNN
Vol. 2 No. 1 2026 pp. 01-10

Spatial Analysis of Residential Areas in Densely Populated Regions of Bireuen City Using K-Medoids Clustering

Ulfi Sahara[1], Muhammad Rizka[2]*, Hendrawaty[3]

[1,2,3] Department of Information and Computer Technology, State Polytechnic of Lhokseumawe
Jln. B.Aceh Medan Km.280 Buketrata 24301 Indonesia

*Corresponding Author: rizka@pnl.ac.id

Article info: Received month dd, yyyy, Revised month dd, yyyy, Accepted month dd, yyyy

Abstract

The rapid population growth in Bireuen City, Aceh, poses significant challenges for urban planning, particularly in managing high population density and various social issues. Based on data from the Central Bureau of Statistics, Aceh's population increased by 2%, reaching 5.37 million people in 2019, with a population density ratio of 95 people per km². Bireuen City, with a population of 471,635, is experiencing rapid growth across diverse occupational backgrounds. This study aims to develop a web-based population density mapping system using the K-Medoids method to overcome the limitations of the K-Means method in handling outliers. The data used include district names, area size, number of ID cards, number of households, and total population. The K-Medoids algorithm groups the data into the nearest clusters, and the process is repeated until the clustering results become stable. The clustering results indicate three main clusters: Cluster 0 (medium density, Silhouette Score 0.70), Cluster 1 (low density, Silhouette Score 0.20), and Cluster 2 (high density, Silhouette Score 1). These findings are expected to assist the Central Bureau of Statistics in planning residential areas and managing urban development more effectively and efficiently.

Keywords: Clustering, Density, Area, Silhoutte Score, K-Medoids

1. Introduction

The rapid development of information technology has driven various sectors to become more adaptive in utilizing technology for decision-making processes. This technology, encompassing hardware, software, and infrastructure, enables different sectors to manage, access, and organize data more efficiently. One of the sectors significantly affected by this advancement is the housing development sector, particularly in planning areas with high population density [1][2].

Bireuen City, located in Aceh Province, Indonesia, is one of the cities experiencing continuous population growth. However, this growth has not been accompanied by adequate housing planning, which has ultimately led to various social problems such as slum settlements, excessive population density, and a decline in residents' quality of life [3]. This situation compels the city government, together with the Central Bureau of Statistics (BPS), to find solutions for mapping and clustering regions based on population density levels to optimize housing development.

To address this issue, a system was developed titled "Application of K-Medoids Clustering for Determining Residential Areas in Densely Populated Regions of Bireuen City." One of the methods suitable for solving this problem is the K-Medoids Clustering method [4]. This method enables the integration of uncertain variables such as total area, number of households, number of ID cards, total population, and number of settlements with greater accuracy [5]. It is capable of grouping data based on specific similarities, which in this study are applied to population data across districts in Bireuen City. The resulting clusters provide valuable information regarding regions that require housing development based on population density categories—high, medium, and low [6]. With this system, it is expected that more accurate mapping can be achieved to support decision-making in housing development. This will have a positive impact on the quality of life of Bireuen's residents while contributing to resolving population density issues and ensuring a more efficient and effective distribution of housing development.

2. Methods

A. System Requirements Analysis

The system requirements analysis in this study is divided into two categories: functional and non-functional requirements. The functional requirements analysis relates to the features or functionalities that the system must possess to meet user needs. On the other hand, the non-functional requirements analysis concerns system attributes that are not directly related to its core functions but rather to the overall quality and characteristics of the system.

1) Functional Requirements Analysis

Functional requirements include all entities involved in the system along with the processes performed by these entities. The functional requirements that can be carried out by the admin in this system are as follows:

- Logging into the system
- Logging out of the system
- Inputting datasets into the database through the system
- Inputting district (sub-district) data
- Performing predictions and viewing the results in the system
- Viewing, editing, and deleting datasets in the database through the system
- Viewing prediction result graphs in the system
- Viewing K-Medoids clustering results in the system

2) Non-Functional Requirements Analysis

Non-functional requirements consist of hardware and software components. Hardware refers to the physical devices that can be seen and touched, while software refers to the programs or applications that cannot be physically touched. The non-functional requirements of this system are as follows:

a. Hardware

The hardware used in the development of this thesis consists of a single laptop utilized to complete the entire process of system development and research reporting. The hardware specifications used are as follows: an **ASUS Vivobook X415DAP** laptop equipped with an **AMD Ryzen 3 3250U CPU @ 2.6GHz**.

b. Software

The software specifications used in developing this system are as follows:

- Operating System: Windows 10
- Web Browser: Google Chrome
- Diagram and UI Design Tools: Draw.io and Figma
- Server Application: XAMPP
- Code Editor: Visual Studio Code version 1.89.1
- Programming Language: JavaScript

B. Data Collection

Data refers to a collection of unprocessed facts that have not yet produced any information. When designing a system, relevant and reliable data is essential. The data used in this study consists of a population density dataset obtained from the Bireuen Central Bureau of Statistics (BPS). The dataset includes information such as district names, total area size, number of ID cards (KTP), number of households (KK), and number of settlements. The data collected covers the years 2019 to 2023.

C. Data Requirements Analysis

The data requirements analysis in this study includes population density data for Bireuen Regency.

D. System Design

The system design in this study includes the development of a Use Case Diagram and an Activity Diagram. This design aims to provide an overview of the system's workflow in determining and grouping areas based on population density using the K-Medoids method, implemented in a web-based system for the Bireuen Central Bureau of Statistics.

1) Use Case Diagram

The design of the use case diagram in this study involves a single actor, namely the Admin of the Bireuen Central Bureau of Statistics (BPS). This actor serves as the main user who has access to all system functionalities. The admin can perform various actions and interact with the system according to operational needs and objectives. The use case diagram design of this study is illustrated in Figure 1.

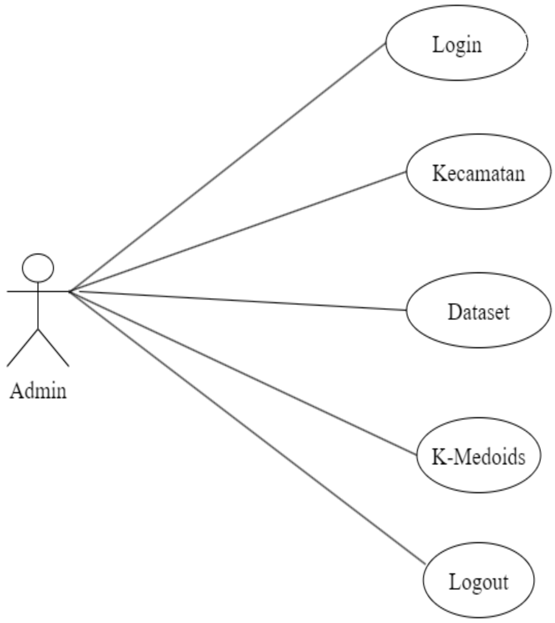

Figure 1 Use Case Diagram

E. K-Medoids Clustering Process Design

The stages of the K-Medoids clustering process are illustrated in the flowchart shown in Figure 2.

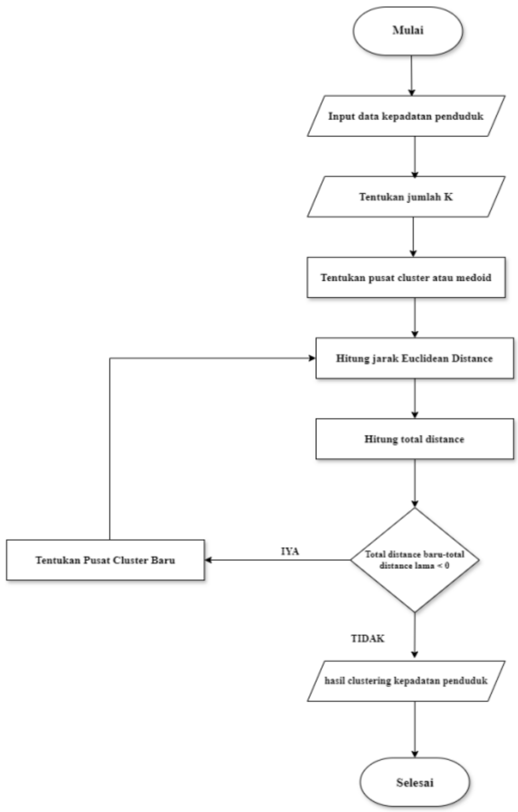

Figure 2. Stages of the K-Medoids Clustering Flowchart

1. The system begins by inputting data such as district name, total area, number of ID cards (KTP), number of households (KK), total population, and number of settlements.
2. Determine the number of clusters (k) to be formed.
3. Select k initial cluster centers (medoids) randomly.
4. Then, assign each data point (object) to the nearest cluster using the appropriate distance formula.
5. After that, reassign each data point to the nearest cluster until the cluster membership remains stable.

$$d(x,y) = ||x - y||$$

$$d(x,y) = \sqrt{\sum_{i=1}^{n} \quad (Xi - Yi); 1,2,3, \ldots . n(1)}$$

6. Next, calculate the total distance for all data points within their respective clusters.
7. After obtaining the total distance, calculate the deviation (S), which is the difference between the new total distance and the previous total distance.
8. If S < 0, the process is repeated from steps 4 to 6 until the deviation value becomes greater than 0.
9. If S ≥ 0, the clustering process is stopped, and the final clusters for each medoid are obtained.
10. The workflow is completed when the system displays the final output, showing clusters categorized as high, medium, and low population density.

F. Database Design

The database designed for this population density system serves as a storage medium for data entered by the admin, such as data grouping based on population density levels. The database used in this system is structured to store various types of data divided into three main tables: the account table, dataset table, and district data table. The database management system (DBMS) used in this study is MySQL.

G. User Interface Design

The user interface design of this system aims to provide a clear and intuitive guide for users in operating the system effectively. The interface components are designed to meet the needs of the admin, ensuring ease of navigation and usability. The system's user interface includes several main pages, such as Home Page, Login Page, Dashboard Page, District (Kecamatan) Page, Dataset Page, Add Dataset Page, and K-Medoids Page.

3. Result and Discussions

A. User Interface

1. Login Page Display

The login form page serves as the initial interface before accessing the system. This page allows the admin to enter their credentials to gain authorized access to the system. The display of the login form page is shown in Figure 3.

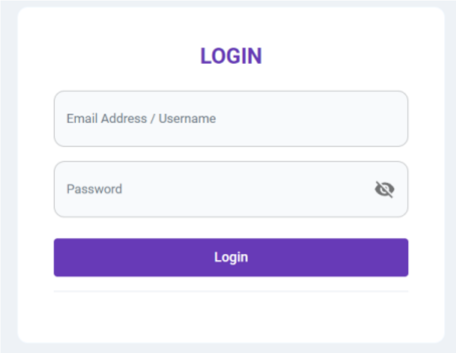

Figure 3 Login Page

2. Dashboard Page Display

The dashboard page is the first page accessible to the user after successfully logging into the system. This page presents several pieces of information related to the system, such as the total number of data entries used, details regarding population density information, and other relevant system insights. The display of the dashboard page is shown in Figure 4.

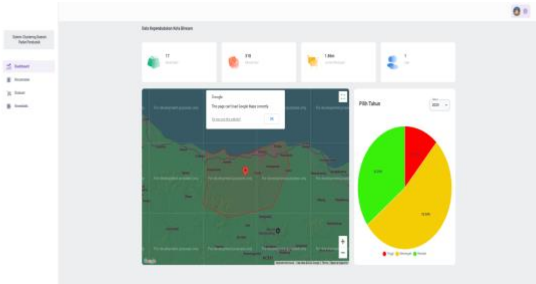

Figure 4 Dashboard Page

3. K-Medoids Page Display

The K-Medoids page is used by the user to perform population density clustering and identify which areas have high population density using the K-Medoids method. This page provides visualization and results of the clustering process. The display of the K-Medoids page is shown in Figure 5.

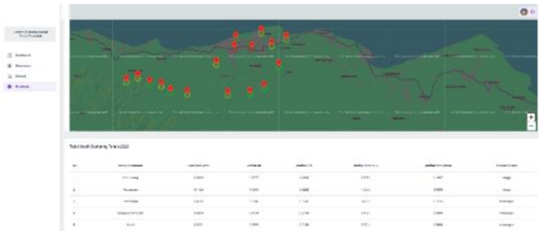

Figure 5. K-Medoids Page Display

B. K-Medoids Calculation

The collected dataset is processed using five variables. The data, obtained from the years 2019 to 2023, serves as the basis for the clustering process. In the K-Medoids method, it is necessary to determine the number of clusters to be used before performing the calculation. The determination of cluster numbers can be seen in Table 1 below.

Table 1. Cluster Label

Klaster	Label
C1	High
C2	Low
C3	Medium

From the entire dataset, the researcher will apply a manual calculation example using the data from 2019. The criteria for the processed data are determined based on the population numbers from 2019 to 2023, as shown in Table 2 below.

Table 2. Dataset 2019

Subdistrict	Area	ID	Resident card	Residents	Residential
Samalanga	141.42	5087	6787	27901	5
Simpang Mamplam	155.5	4692	6653	27237	3
Pandrah	114.01	5414	6849	8806	3
Jeunib	112.51	6685	9035	25244	6
Peulimbang	127.89	10926	12718	12078	3
Peudada	312.81	2357	3756	27839	6
Juli	231.18	4512	6361	33731	4
Jeumpa	109.04	1534	2084	36930	5
Kota juang	16.91	7347	8426	47670	4
Kuala	17.11	4426	6701	18731	4
Jangka	37.58	4436	5803	28687	5
Peusangan	59.08	2067	2718	52716	9
Peusangan selatan	94.15	14739	11732	14889	3
Peusangan siblah krueng	111.57	5456	4424	11956	3
Makmur	71.74	1972	3261	15656	4
Gandapura	46.95	3006	3570	23731	4
Kutablang	38.7	4087	5545	22602	4

The steps involved in performing the K-Medoids calculation are as follows:

1. The number of clusters used in grouping densely populated areas is three (3) clusters. These clusters are categorized as High (C1), Medium (C2), and Low (C3) population density.
2. The purpose of data normalization in the clustering process is to adjust the feature values so that they are on the same scale. This step is important because, without normalization, features with larger numeric ranges may dominate those with smaller ranges, potentially affecting the accuracy of the clustering results. The normalization function used is Min-Max Scaling, and the results are presented as follows:

   $x1y1$=(141.42—16.91)/(312.81—16.91)=0,4028
   $x2y1$=(5087—16.91)/(312.81—16.91)=0,2691
   $x3y1$=(6787—16.91)/(312.81—16.91)=0,4423
   $x4y1$=(27901—16.91)/(312.81—16.91)=0,4349
   $x5y1$=(5—16.91)/ (312.81—16.91)=0,3333

Table 3. Data Normalization

Subdistrict	Area	ID	Resident card	Residents	Residential
Samalanga	0.4208	0.2691	0.4423	0.4349	0.3333
Simpang Mamplam	0.4684	0.2392	0.4297	0.4971	0
Pandrah	0.3282	0.2938	0.4481	0	0
Jeunib	0.3231	0.3901	0.6537	0.3744	0.5
Peulimbang	0.3751	0.7112	1	0.0745	0
Peudada	1	0.0623	0.1572	0.4335	0.5
Juli	0.7241	0.2255	0.4022	0.5676	0.1667
Jeumpa	0.3114	0	0	0.6405	0.3333
Kota juang	0	0.4402	0.5964	0.8851	0.1667
Kuala	0.0007	0.2190	0.4342	0.2260	0.1667
Jangka	0.0699	0.2198	0.3497	0.4528	0.3333
Peusangan	0.1425	0.0404	0.0596	1	1
Peusangan selatan	0.2610	1	0.9073	0.1385	0
Peusangan siblah krueng	0.3199	0.2970	0.2200	0.0717	0
Makmur	0.1853	0.0332	0.1107	0.1560	0.1667
Gandapura	0.1015	0.1115	0.1397	0.3399	0.1667
Kutablang	0.0736	0.1933	0.3255	0.3142	0.1667

3. The initial cluster centers (Medoids) are selected randomly. These medoids serve as the starting points for the clustering process. The initial medoids used in this calculation are shown in Table 4 below.

Table 4. Initiation Medoid

	Initial Medoid				
C1	0.4208	0.2691	0.4423	0.4349	0.3333
C2	1	0.0623	0.1572	0.4335	0.5
C3	0.1425	0.0404	0.0596	1	1

4. Calculate the Euclidean distance to the predetermined initial medoids.

$$d(x) = \sqrt{(0.4208 - 0.4208)^2 + (0.2691 - 0.2691)^2 + (0.4423 - 0.4423)^2 + (0.4349 - 0.4349)^2 + (0.333 - 0.333)^2}$$

$$d(x) = \sqrt{(1 - 0.4208)^2 + (0.0623 - 0.2691)^2 + (0.1572 - 0.4423)^2 + (0.4335 - 0.4349)^2 + (0.5 - 0.333)^2}$$

$$d(x) = \sqrt{(0.1425 - 0.4208)^2 + (0.0404 - 0.2691)^2 + (0.0596 - 0.4423)^2 + (1 - 0.4349)^2 + (1 - 0.333)^2}$$

Continue the process for all remaining data rows by calculating the Euclidean distance in the same manner as before. After obtaining all Euclidean distances, select the smallest distance value for each data point. The total distance for all data within each cluster can be seen in Table 5.

Table 5. Euclidean Distance For First Iteration

No	C1	C2	C3	Distance
1	5.7803	0.6980	1.0197	5.7804
2	0.3359	0.7989	1.2725	0.3359
3	0.5562	1.0136	1.4998	0.5562
4	0.3167	0.9030	1.0720	0.3167
5	0.8658	1.3786	1.8013	0.8658
6	0.6980	1.3786	1.1472	0.6980
7	0.3753	0.5402	1.1710	0.3753
8	0.5676	0.7572	0.7793	0.5676
9	0.6786	1.2848	1.0844	0.6786
10	0.5004	1.1198	1.2189	0.5004
11	0.3666	0.9777	0.9303	0.3666
12	1.019	1.1472	3.4997	3.4997
13	0.9874	1.5247	1.8427	0.9874
14	0.5507	0.9499	1.4088	0.5507
15	0.5714	0.9245	1.2940	0.5714
16	0.5050	0.9642	1.0692	0.5050
17	0.4268	1.014	1.1241	0.4268
Total jarak				6.2284

5. New medoids are selected randomly, with the condition that previously used medoids cannot be chosen again. The newly selected medoids can be seen in Table 6 below.

Table 6. New Medoids

New Medoids					
C1	0	0.4402	0.5964	0.8851	0.1667
C2	0.0007	0.2190	0.4342	0.2260	0.1667
C3	0.0699	0.2198	0.3497	0.4528	0.3333

6. After obtaining the new medoids, the data is recalculated using the Euclidean distance with the new medoid centers. The calculation process is the same as in the first iteration. The results of the second iteration can be seen in Table 7.

Table 7. Second Iteration

No	C1	C2	C3	Distance
1	0.6786	0.5004	0.3666	0.3666
2	0.7293	0.5333	0.5269	0.5269
3	0.9809	0.4380	0.4380	0.4380
4	0.6943	0.5607	0.8679	0.4684
5	1.0304	0.8679	0.9772	0.8679
6	1.2848	1.1198	0.6868	0.9772
7	0.8419	0.8000	0.5139	0.6868
8	0.9901	0.7297	4.0782	0.5139
9	4.0783	0.7138	5.8254	4.0782
10	0.7138	5.8253	7.8774	5.8254
11	0.5735	0.3018	0.9303	7.8774
12	1.0843	1.2189	1.0789	0.9303
13	1.0311	0.9679	0.5844	0.9679
14	0.9766	0.4532	0.4702	0.4532
15	0.9836	0.4221	0.3119	0.4221
16	0.7900	0.3484	0.2197	0.3119
17	0.6824	0.1599	0.6308	0.1599

7. After obtaining the distance calculation results in the second iteration, the next step is to calculate the total deviation (S) to determine whether the iteration process should continue or stop. The deviation value is calculated using the formula $S = \text{Total New Distance} - \text{Total Previous Distance}$ S=Total New Distance−Total Previous Distance. Based on the results obtained, the total new distance is 8.0918, and the total previous distance is 6.2284, resulting in a deviation value of $S = 8.0918 - 6.2284 = 1.8634$ S = 8.0918−6.2284=1.8634.
   Since the deviation value is greater than zero (S > 0), it can be concluded that the iteration process does not need to continue, as no improvement occurs in the clustering results. Therefore, the computation stops at the second iteration, and the results from this final iteration are used as the final clustering parameters. The final clustering results can be seen in Table 8 below.

Table 8. Final Cluster Results

NO	C1	C2	C3	Distance	Cluster	Results
1	0.6786	0.5004	0.3666	0.3666	C1	High
2	0.7293	0.5333	0.5269	0.5269	C1	High
3	0.9809	0.4380	0.4380	0.4380	C2	Medium
4	0.6943	0.5607	0.8679	0.4684	C2	Medium
5	1.0304	0.8679	0.9772	0.8679	C2	Medium
6	1.2848	1.1198	0.6868	0.9772	C2	Medium
7	0.8419	0.8000	0.5139	0.6868	C2	Medium
8	0.9901	0.7297	4.0782	0.5139	C2	Medium
9	4.0783	0.7138	5.8254	4.0782	C3	Low
10	0.7138	5.8253	7.8774	5.8254	C3	Low
11	0.5735	0.3018	0.9303	7.8774	C3	Low
12	1.0843	1.2189	1.0789	0.9303	C3	Low
13	1.0311	0.9679	0.5844	0.9679	C3	Low
14	0.9766	0.4532	0.4702	0.4532	C3	Low
15	0.9836	0.4221	0.3119	0.4221	C3	Low
16	0.7900	0.3484	0.2197	0.3119	C3	Low
17	0.6824	0.1599	0.6308	0.1599	C3	Low

C. Final Clustering Results

1. The results of the calculations using the K-Medoids method on the 2020 dataset can be seen in Table 9 below.

Table 9. Dataset 2020

No	Cluster	Number of Subdistict	Percentage(%)
1	High (C1)	2	11,76%
2	Medium (C2)	9	35,29%
3	Low (C3)	6	52,94%

The results are visualized using a bar chart, as shown in Figure 6 below.

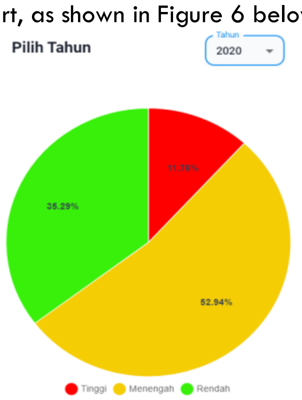

Figure 6 Percentage Chart for the Year 2020

2. The results of the calculation using the K-Medoids method for the 2021 data can be seen in Table 10 below.

Table 10. Dataset 2021

No	Cluster	Number of Subdistict	Percentage(%)
1	High (C1)	5	29,41%
2	Medium (C2)	7	41,18%
3	Low (C3)	5	29,41%

The results are visualized using a bar chart, as shown in Figure 7 below.

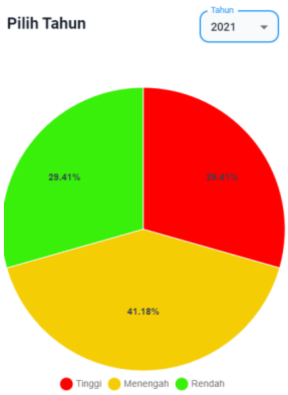

Figure 7 Percentage Chart for the Year 2021

3. The results obtained from the K-Medoids clustering method for the 2022 dataset can be seen in Table 11.

Table 11. Dataset 2022

No	Cluster	Number of Subdistict	Percentage(%)
1	High (C1)	8	47,06%
2	Medium (C2)	3	17,65%
3	Low (C3)	6	35,29%

The results are visualized using a bar chart, as shown in Figure 8.

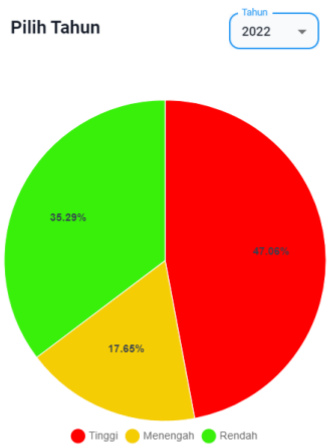

Figure 8 Percentage Chart for the Year 2022

4. The results of the calculation using the K-Medoids method for the 2023 data can be seen in Table 12 below.

Table 12. Dataset 2023

No	Cluster	Number of Subdisctrict	Percentage (%)
1	High (C1)	1	5,88%
2	Medium (C2)	9	52,94%
3	Low (C3)	7	41,18%

The results are visualized using a bar chart, as shown in Figure 9.

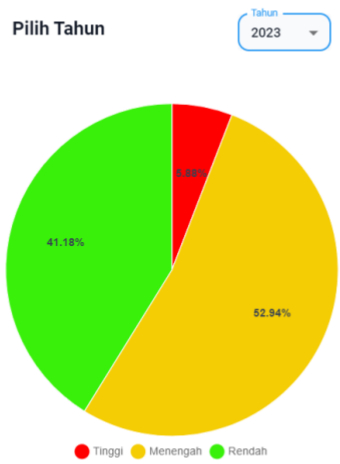

Figure 9. Percentage Chart for the Year 2022

D. Testing

After applying the K-Medoids method to the dataset, the clustering quality test was carried out using the Silhouette Score. The calculation of the Silhouette Score produced an average value of X, indicating that the clustering quality was Y. A cluster with a value close to 1 shows that the data within the cluster is well-grouped, while a cluster with a value close to 0 or negative indicates the possibility of data being incorrectly assigned to a cluster. The detailed results can be seen in Table 13 below.

Table 13. Silhouette Score Testing

No	Cluster	Shilhoutte Score	Information
1	Cluster 0	0.70	Indicates that the members in this cluster have a fairly good level of closeness.
2	Cluster 1	0.20	Indicates that members in this cluster have a better level of closeness with other members of their own cluster than with members of other clusters.
3	Cluster 2	1	Indicates that members in this cluster have a very good level of closeness with other members of the same cluster, and are relatively more distant from other members of the same cluster. Overall, the silhouette score results indicate that cluster 2 has the best separation.

4. Conclusion

Based on the evaluation of the research objectives, namely developing a web-based system to cluster densely populated areas using K-Medoids clustering, several key conclusions can be drawn. First, the designed web-based system was successfully built and functions optimally in identifying and classifying regions according to population density levels. This system facilitates users in analyzing the distribution of population density across various subdistricts in Bireuen City. Second, the K-Medoids algorithm performs well in grouping the data into three density clusters: high, medium, and low. This process involves the random initialization of medoids, calculating the Euclidean distance between the data and the nearest medoid, and updating the medoids based on the minimized total distance, which is repeated until the medoids no longer change or the maximum number of iterations is reached. Third, the implementation of the K-Medoids method produces valid clustering, with Silhouette Score values of 0.70 for the medium-density cluster, 0.20 for the low-density cluster, and 1.00 for the high-density cluster. These values provide a clear overview of population density distribution, which can support more effective housing planning in the city.

REFERENCES

[1] N. Nabiilah, F. J. Fauzan, N. Nurazizah, et al., “Application of K-Means and K-Medoids Algorithms in Clustering of Densely Populated Areas in Riau Province: Penerapan Algoritma K-Means dan K-Medoids Dalam …,” … Penelit. dan …, pp. 223–229, 2023. [Online]. Available: https://journal.irpi.or.id/index.php/sentimas/article/view/561

[2] Sumiati, A. Rasyid, F. Adinata, et al., “Penerapan Metode Two-Step Cluster Untuk Pengelompokan Desa Berdasarkan Kepadatan Penduduk,” vol. 05, no. 02, pp. 36–41, 2021.

[3] P. Marpaung, R. F. Siahaan, “Penerapan Algoritma K-Means Clustering Untuk Pengelompokan Daerah Berdasarkan Kepadatan Penduduk,” J. Sains Komput. Inform. (J-SAKTI), vol. 5, no. 1, pp. 503–521, 2021.

[4] B. Wira, A. E. Budianto, and A. S. Wiguna, “Implementasi Metode K-Medoids Clustering Untuk Mengetahui Pola Pemilihan Program Studi Mahasiswa Baru Tahun 2018 Di Universitas Kanjuruhan Malang,” RAINSTEK J. Terap. Sains Teknol., vol. 1, no. 3, pp. 53–68, 2019, doi: 10.21067/jtst.v1i3.3046.

[5] S. Velamparambil, S. Mackinnon-Cormier, J. Perry, et al., “Algoritma K-Medoids,” Proc. 38th Eur. Microw. Conf. EuMC 2008, vol. 1, no. 9, pp. 1312–1314, 2008, doi: 10.1109/EUMC.2008.4751704.

[6] L. Maulida, “Penerapan Datamining Dalam Mengelompokkan Kunjungan Wisatawan Ke Objek Wisata Unggulan Di Prov. DKI Jakarta Dengan K-Means,” JISKA (Jurnal Inform. Sunan Kalijaga), vol. 2, no. 3, p. 167, 2018, doi: 10.14421/jiska.2018.23-06.

[7] J. B. MacQueen, “Some Methods for Classification and Analysis of Multivariate Observations,” Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, vol. 1, no. 14, pp. 281–297, 1967.

[8] A. K. Jain, M. N. Murty, and P. J. Flynn, “Data Clustering: A Review,” ACM Computing Surveys, vol. 31, no. 3, pp. 264–323, 1999, doi: 10.1145/331499.331504.

[9] G. Gan, C. Ma, and J. Wu, Data Clustering: Theory, Algorithms, and Applications. Society for Industrial and Applied Mathematics, 2007.

[10] R. Xu and D. C. Wunsch, “Survey of Clustering Algorithms,” IEEE Transactions on Neural Networks, vol. 16, no. 3, pp. 645–678, 2005, doi: 10.1109/TNN.2005.845141.