



Implementasi Algoritma K-Nearest Neighbor (KNN) Untuk Prediksi Penyakit Diabetes Berbasis Web

Heri Setiawan Fauzi^a, Khanif Faozi^b

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Pamulang, Tangerang Selatan, Indonesia

Article Info

Keywords:

K-Nearest Neighbor
Diabetes Mellitus
Web-Based Prediction
Early Detection
Machine Learning

ABSTRACT

Diabetes mellitus is a chronic metabolic disease that requires early detection to reduce the risk of serious complications. Limited access to practical screening tools encourages the development of a web-based prediction system that can be accessed independently. This study develops a web application for predicting diabetes risk using the K-Nearest Neighbor (KNN) classification algorithm. The system processes health parameters consisting of glucose level, blood pressure, body mass index (BMI), age, and gender. System development applies the Software Development Life Cycle (SDLC) Waterfall model, covering requirements analysis, system design, development, testing, and maintenance. The developed application provides data management, K-value configuration, prediction, and prediction-history features. Black Box testing shows that the implemented functions operate according to the expected scenarios. Accuracy testing using 10 test data produced 10 correct predictions and no incorrect predictions, resulting in an accuracy of 100%. The results indicate that the developed application can support early diabetes-risk screening and demonstrate that KNN can be integrated into a web-based prediction system.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Heri Setiawan Fauzia
Program Studi Teknik Informatika
Universitas Pamulang
Tangerang Selatan, Indonesia
Email: herise23@gmail.com

1. INTRODUCTION

Diabetes mellitus is one of the chronic diseases that requires early identification because uncontrolled high blood glucose can lead to serious complications. The development of information and communication technology provides opportunities to support health services through systems that can process data and present information quickly. A web-based approach can make an initial screening tool easier to access without requiring users to go through a complicated procedure at the early stage [2], [25].

Machine learning can be used to classify health data based on patterns contained in training data. K-Nearest Neighbor (KNN) is a classification method that determines the class of a new record by comparing its distance to existing training records. Previous studies have applied KNN to diabetes prediction and reported that the method can provide useful classification performance when supported by appropriate data preparation and parameter selection [4], [13], [16], [22]. Other comparative research has also shown that the performance of KNN needs to be evaluated carefully against alternative classification approaches [7].

Previous web-based diabetes studies have generally focused on different datasets, preprocessing techniques, interfaces, or optimization strategies. For example, Ruziq and Wayahdi used the Early Stage Diabetes Risk Prediction Dataset and tested several K values, while Riansyah and Syahputra developed a web-based diabetes classification system using the Pima Indians Diabetes Database [4], [16]. Arfiah et al. focused on optimization using preprocessing, hyperparameter tuning, and SMOTE [22]. These studies provide a basis for developing a practical web-based prediction application using KNN.

Based on this background, this research focuses on developing a web-based diabetes prediction application that processes glucose, blood pressure, BMI, age, and gender as input attributes. The research uses the Waterfall SDLC model and evaluates the application using functional testing and an accuracy test. The contribution of this research is the integration of KNN classification into a web application with dataset management, K-value configuration, prediction, and prediction-history functions so that early screening can be performed more practically.

2. RESEARCH METHOD

This research follows the SDLC Waterfall model. The stages are requirement analysis, system and software design, development, integration and system testing, and operation and maintenance. The stages are performed sequentially so that the output of one stage becomes the basis for the next stage. The Waterfall approach is also described in the thesis as a structured development model covering requirements, design, coding, testing, and maintenance [24].

Data collection was conducted through observation and literature study. The research uses a diabetes dataset with clinical attributes including glucose, blood pressure, BMI, age, gender, and a diabetes-status label. The thesis states that the dataset is associated with the Pima Indians Diabetes Database and that the system uses the selected health parameters for classification. The application was developed as a web system using PHP and MySQL.

KNN classification determines the class of a new data record from the nearest training records. The distance between a test record and a training record is calculated using Euclidean distance. For two records x and y with n attributes, the distance can be expressed as:

$$d(x,y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (1)$$

The system then selects the nearest records according to the configured K value and determines the predicted class from the majority class among those neighbors. The application provides an administrator with facilities for managing users, training data, and the K value, while the prediction process accepts the required health parameters and stores the prediction result in the history.

System evaluation consists of functional testing using Black Box scenarios and accuracy testing. Black Box testing checks whether each feature produces the expected output. Accuracy testing compares the KNN prediction with the actual label of the test data using the following equation:

$$\text{Accuracy} = (\text{Number of correct predictions} / \text{Total test data}) \times 100\% \quad (2)$$



Figure 1. Waterfall software development model

Figure 1 summarizes the sequential development process used in this research, beginning with requirements analysis and continuing through design, development, testing, and maintenance.

2.1 RESEARCH REQUIREMENTS AND PROCESS

The research requirements were derived from the problem of providing a practical web-based tool for initial diabetes-risk screening. The system is designed to accept clinical parameters that are available in the dataset and commonly used in the developed application, namely glucose, blood pressure, body mass index (BMI), age, and gender. The prediction result is presented as a classification of Diabetes or Normal. In addition to the prediction function, the application provides supporting functions for authentication, user management,

training-data management, K-value configuration, and prediction history. These functions are included so that the classification process can be managed through an integrated application rather than through a standalone calculation.

The research process begins with requirement analysis and data collection through observation and literature study. The next stage is system and software design, followed by coding and integration. After implementation, functional testing is performed using Black Box scenarios and the prediction capability is evaluated using a separate accuracy test. The final stage is operation and maintenance. This sequence follows the Waterfall model described in the thesis and provides a clear relationship between the identified requirements, system implementation, and evaluation activities [24].

2.2 DATASET AND INPUT ATTRIBUTES

The dataset used in this research contains health-related attributes and a class label for diabetes status. The thesis associates the dataset with the Pima Indians Diabetes Database and uses selected attributes for the classification process. In the implemented application, the prediction form uses glucose, blood pressure, BMI, age, and gender as input variables. The class label is used as the reference for evaluating whether the predicted result corresponds to the known status of the test record.

No.	Attribute	Role in Prediction	Description
1	Glucose	Input	Blood glucose value used as one of the main classification attributes.
2	Blood pressure	Input	Blood pressure value included in the patient record.
3	BMI	Input	Body mass index value representing the body-mass characteristic.
4	Age	Input	Patient age used as a classification attribute.
5	Gender	Input	Patient gender recorded as part of the prediction input.
6	Diabetes status	Class label	Reference class used to compare the KNN prediction with the known result.

Table 1. Input attributes used by the web-based prediction system

The use of several attributes allows the system to classify a new record based on a combination of health characteristics rather than relying on a single variable. In the KNN process, the new record is compared with the available training records. The resulting similarity is determined from the distance between attribute values, while the known class of the nearest records becomes the basis for the final classification.

2.3 K-NEAREST NEIGHBOR CLASSIFICATION PROCEDURE

K-Nearest Neighbor is applied as a classification method in which a new data record is compared with existing training records. The first step is to determine the K value that represents the number of nearest records considered. The system then calculates the distance between the new record and the training data, sorts the distances from the smallest to the largest, selects the K nearest records, and determines the class based on the majority class among those neighbors. This procedure follows the KNN principle described in the thesis and is implemented as the core prediction process.

The distance calculation uses Euclidean Distance. For a test record x and a training record y with n attributes, the distance can be expressed as:

$$d(x,y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

where $d(x,y)$ represents the distance between the test and training records, x_i represents the value of an attribute in the test record, y_i represents the corresponding value in the training record, and n represents the number of attributes used. After all distances are calculated, the records are sorted and the nearest K records are selected. The majority class among these neighbors is then used as the prediction result. The application also provides an interface for configuring the K value so that the classification parameter can be managed through the system.

2.4 SYSTEM DESIGN AND IMPLEMENTATION

The system is implemented as a web application using PHP and MySQL. The application separates the management functions from the prediction process. Authentication is used to control access to the application, while the dataset management function provides an interface for maintaining the training records. The K-value setting stores the parameter used during classification. The prediction page accepts the

required health attributes and displays the resulting class, while the prediction-history function stores and presents previous prediction records.

The implementation follows the system design described in the thesis. The web interface is arranged so that the main functions can be reached through the application navigation. This structure supports the operational flow from entering data, processing the KNN calculation, obtaining a prediction, and reviewing the resulting history. The use of a database also allows training data and prediction records to be stored so that they can be managed through the application interface.

2.5 EVALUATION PROCEDURE

Evaluation is divided into functional testing and accuracy testing. Functional testing uses Black Box scenarios, meaning that the evaluation focuses on the behavior observed from each function without examining the internal implementation of the source code. The scenarios cover login and logout, user management, training-data management, K-value configuration, diabetes prediction, and prediction history. Each scenario is considered valid when the actual system behavior is consistent with the expected result.

Accuracy testing is performed separately using ten test records. Each record contains glucose, blood pressure, BMI, age, and gender, together with its known dataset label. The system prediction is compared with the reference label. The number of correct predictions is then divided by the total number of test records and multiplied by 100 percent. This procedure is used to determine the accuracy reported in the research.

3. RESULTS AND DISCUSSION

3.1 SYSTEM IMPLEMENTATION

The implementation produced a web-based diabetes prediction application consisting of several connected pages. The login page functions as the entry point to the application. After successful authentication, the user can access the dashboard and the available management and prediction functions according to the implemented access flow. The application was developed to support both the administration of the training data and the execution of predictions from new patient records.

Figure 2. User management page

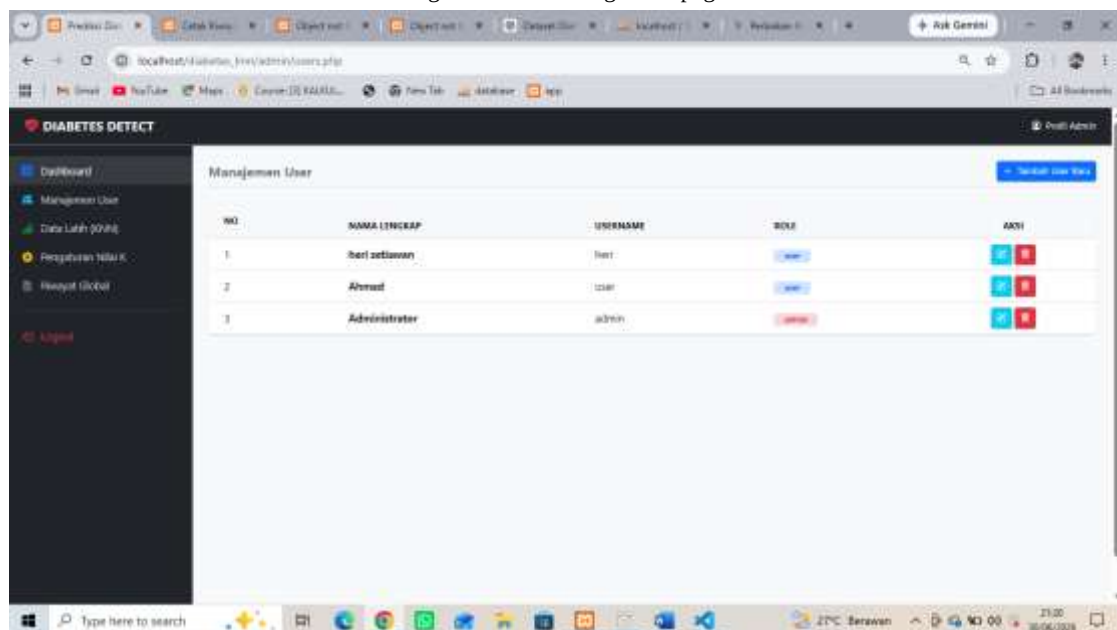


Figure 2 shows the user management interface used to maintain application users. The available operations include adding a new user, changing user information, and deleting user data. The management function is important because the system requires an authenticated user before the prediction and administrative functions can be accessed. The corresponding Black Box scenarios verify that valid data can be stored and that incomplete required fields are rejected.

Figure 3. Training-data management page

NO	GLUKOSA	TEKANAN DARAH	BMI	USIA	GENDER	STATUS / LABEL	AKSI
1	103 mg/dL	71 mmHg	22.6	28 Thn	L	Normal	[Edit] [Delete]
2	89 mg/dL	65 mmHg	20.4	21 Thn	P	Normal	[Edit] [Delete]
3	115 mg/dL	76 mmHg	24.5	36 Thn	P	Normal	[Edit] [Delete]
4	108 mg/dL	72 mmHg	23.2	30 Thn	L	Normal	[Edit] [Delete]
5	96 mg/dL	70 mmHg	21.7	23 Thn	P	Normal	[Edit] [Delete]
6	201 mg/dL	91 mmHg	35.2	59 Thn	P	Diabetes	[Edit] [Delete]
7	174 mg/dL	84 mmHg	31.6	44 Thn	L	Diabetes	[Edit] [Delete]
8	192 mg/dL	89 mmHg	34.0	53 Thn	P	Diabetes	[Edit] [Delete]

Figure 3 presents the training-data management interface. The page is used to maintain the dataset that becomes the reference for KNN classification. The testing process verifies the main CRUD operations, including adding, editing, and deleting training records. Validation is also performed by submitting incomplete data to ensure that the application does not save records when required fields have not been completed.

Figure 4. Prediction and history interface

NO	TANGGAL CEK	NAMA PASIEN	GLUKOSA	BMI	HASIL	AKSI
1	30/06/2026 20:56	heri	2010 mg/dL	33.0	Diabetes	[Delete]
2	30/06/2026 20:55	heri	98 mg/dL	30.0	Normal	[Delete]
3	23/05/2026 00:23	user	110 mg/dL	22.0	Normal	[Delete]
4	16/05/2026 12:58	user	60 mg/dL	15.0	Normal	[Delete]
5	16/05/2026 12:57	user	60 mg/dL	10.0	Normal	[Delete]
6	16/05/2026 12:56	user	70 mg/dL	10.0	Normal	[Delete]
7	16/05/2026 12:55	user	120 mg/dL	22.0	Normal	[Delete]

Figure 4 illustrates the prediction-related interface used by the application. The prediction process requires the user to complete the available health parameters before the system performs the KNN calculation. The result can then be reviewed through the prediction interface and stored in the prediction history. The history function provides a record of previous predictions and supports deletion of a selected history record when required.

The resulting application is a web-based system that supports the management of training data and the execution of diabetes prediction. The implementation follows the functional requirements described in the thesis. The main functions include login, dashboard access, user management, KNN training-data management, K-value configuration, prediction, and prediction-history management.

Figure 5. Web-based application login page

Figure 5 shows the initial login interface. After valid authentication, the user is directed to the dashboard. The dashboard provides access to the functions available according to the user's role.

Figure 6. Application dashboard

The dashboard in Figure 6 presents the main system information and navigation. The application also provides a dataset management page so that the administrator can add, edit, and delete training records used by the KNN process. The K-value setting is used to determine the number of nearest neighbors considered during classification.

Figure 7. KNN-related system interface

Figure 7 shows the KNN-related application interface used to support the configuration and execution of the prediction process. The interface connects the configured K value with the data processing stage so that the prediction can be performed using the training records stored by the system.

3.2 DETAILED FUNCTIONAL TESTING

The Black Box evaluation covered 27 scenarios across six functional areas. All scenarios produced results that matched the expected behavior. The detailed scenarios are summarized in Table 4. The result indicates that the application can accept valid input, reject incomplete or invalid input in the tested cases, execute the requested operations, and display the corresponding output.

Function	No.	Test scenario	Expected result	Conclusion
Login/Logout	1	Open the system through the URL	Login page is displayed	Valid
Login/Logout	2	Enter valid username and password	User logs in and reaches the appropriate dashboard	Valid
Login/Logout	3	Enter incorrect username or password	Error message is displayed	Valid
Login/Logout	4	Press Login with empty credentials	Login is rejected and input is requested	Valid
Login/Logout	5	Press Logout	Session ends and Login page is displayed	Valid
User Management	1	Click Add New User	Add-user form is displayed	Valid
User Management	2	Enter complete user data	User data is saved	Valid
User Management	3	Leave a required field empty	Saving is rejected	Valid
User Management	4	Edit user data	User data is updated	Valid
User Management	5	Delete user data	User data is removed from the database	Valid
Training Data	1	Click Add Training Data	Dataset input form is displayed	Valid
Training Data	2	Enter complete dataset data	Training record is saved	Valid
Training Data	3	Leave a field empty	Saving is rejected	Valid
Training Data	4	Edit a dataset record	Training record is updated	Valid
Training Data	5	Delete a dataset record	Training record is deleted	Valid
K Setting	1	Open K-value setting	K setting page is displayed	Valid
K Setting	2	Enter a valid numeric K value	K value is saved	Valid
K Setting	3	Leave K value empty	Saving is rejected	Valid
K Setting	4	Enter letters as K value	Input is rejected	Valid
Prediction	1	Enter all patient data correctly	Diabetes prediction is displayed	Valid
Prediction	2	Leave one examination field empty	System requests completion of the data	Valid
Prediction	3	Press Prediction	KNN calculation is executed and the result is displayed	Valid
Prediction	4	Open prediction details	Classification and nearest-neighbor information are displayed	Valid
Prediction History	1	Open prediction history	History records are displayed	Valid
Prediction History	2	Delete a history record	Selected history record	Valid

			is deleted	
Prediction History	3	Open empty history	System informs the user that no history exists	Valid

Table 2. Detailed Black Box testing results

The detailed test results strengthen the functional evaluation because the validation is not limited to the main prediction button. The login process, database management operations, parameter configuration, prediction input validation, and history management were all tested. The consistent “Valid” result for every scenario shows that the implemented functions behaved according to the requirements tested in the thesis.

3.3 STRUCTURAL TESTING OF CRUD FUNCTIONS

In addition to Black Box testing, the thesis also reports structural testing of the CRUD logic. The tested paths include Create, Read, Update, and Delete operations. The purpose of this evaluation is to verify that the main data-management logic follows the intended execution paths and that the corresponding database operations are triggered by the tested input conditions.

Function	Test path	Test input	Expected output
Create	Node 1 → 3 → 5	Valid save request with POST data	INSERT query succeeds and success status is returned
Read	Node 1 → 5 → 6	Training data with Diabetes status	Record is displayed with the corresponding status indicator
Read	Node 1 → 5 → 6	Training data with Normal status	Record is displayed with the corresponding status indicator
Update	Node 1 → 4 → 5	Update request with an existing ID	UPDATE query succeeds and update status is returned
Delete	Node 1 → 2 → 5	Delete request with an existing ID	DELETE query succeeds and delete status is returned

Table 3. CRUD structural testing summary

The CRUD test results show that the data-management paths used by the application operate as designed in the tested cases. This result is relevant because the KNN prediction depends on the availability of training records. If training data cannot be added, updated, read, or deleted correctly, the prediction process may not operate as intended. Therefore, the CRUD validation forms part of the supporting evaluation of the complete application.

3.4 ACCURACY TESTING AND INTERPRETATION

The accuracy test uses ten records consisting of seven records labeled Diabetes and three records labeled Normal. The system prediction for each record is consistent with the dataset label. The records with higher glucose values in the test set are classified as Diabetes, while the three records labeled Normal are also returned as Normal by the system. The purpose of this test is to compare the classification result generated by the implemented KNN process with the known label in the selected test records.

The accuracy calculation is obtained from ten correct predictions and zero incorrect predictions. Therefore, the accuracy is calculated as follows:

$$\text{Accuracy} = (10 / 10) \times 100\% = 100\%$$

The result indicates that every record included in the selected accuracy test was classified consistently with its reference label. However, the result should be interpreted within the scope of the ten test records used in the research. A 100% result on a small test set does not by itself establish that the system will achieve the same performance on a larger or more diverse population. This limitation is consistent with the thesis recommendation to expand the dataset and conduct further optimization and evaluation.

3.5 DISCUSSION

The first research objective is supported by the availability of a web-based application that enables users to enter health parameters and obtain an initial classification result. The application combines authentication, data management, prediction, and history functions in one environment. This makes the KNN process more accessible than presenting the algorithm only as a calculation procedure. The result also follows the general direction of previous web-based diabetes studies that combine machine-learning classification with an accessible interface [4], [16].

The second research objective concerns the implementation of KNN. The application follows the fundamental KNN sequence: determining K, calculating distance, sorting the distance values, selecting the nearest neighbors, and determining the class from the majority of selected neighbors. Euclidean Distance is used for the distance calculation. The implementation therefore connects the theoretical KNN procedure to a

usable prediction workflow in the web application. Previous studies also indicate that KNN can be applied to diabetes classification, while its performance depends on the data and parameter configuration used [4], [13], [22].

The third research objective concerns effectiveness and accuracy. The reported 100% accuracy demonstrates that the implemented system produced correct classifications for all ten selected test records. The result is useful as an initial verification of the implemented prediction workflow, but it should not be interpreted as clinical diagnostic performance. Further validation using a larger number of test records, more varied patient characteristics, and a more systematic experimental design is required before broader conclusions about model generalization can be made.

The research also has several practical limitations. First, the number of accuracy-test records is limited to ten. Second, the thesis identifies the need for a larger and more varied training dataset. Third, the current implementation uses a configurable K value but does not automatically search for the most optimal value. These limitations provide clear directions for subsequent research. The thesis specifically recommends investigating GridSearchCV for K optimization and SMOTE for handling class imbalance. Such improvements can be evaluated in future work rather than being presented as results of the current system.

From the application perspective, future development can extend the prediction system with additional supporting functions, including health notifications, integration with wearable devices, and mechanisms for follow-up consultation. Because the application processes health-related information, security and database backup should also be strengthened. These points are recommendations for future development and are consistent with the suggestions stated in the thesis.

3.6 Research Implications and Limitations

The results show that the developed application is able to connect the stages of data management, KNN processing, and result presentation in a single web-based workflow. From the user perspective, the process begins with authentication, continues with the management of training data and the K parameter, and ends with a prediction result and a stored history. This integration is important because the usefulness of a classification algorithm in an application is not determined only by the mathematical calculation, but also by whether the surrounding functions can support the data required by that calculation.

From the data-processing perspective, the selected attributes provide the input representation used by the KNN classifier. Glucose, blood pressure, BMI, age, and gender are stored as part of the patient record and are used when a new record is compared with the training data. The Euclidean Distance calculation provides a direct similarity measure between records. The classification result is then determined from the nearest records according to the configured K value. This sequence makes the prediction process traceable because each result is produced from the relationship between a new record and the stored training records.

The functional test results also indicate that the supporting application functions are important to the overall reliability of the system. For example, validation of empty input prevents incomplete records from being stored, while the CRUD test verifies that the training data can be maintained. The K-value setting is also validated so that the application does not accept an empty or nonnumeric value in the tested scenarios. These controls reduce the possibility of operational errors that could interfere with the prediction workflow.

Although the accuracy test produced 100%, the evaluation has an important limitation in the number of records used. Only ten records were used for the accuracy calculation, consisting of ten correct classifications. A small test set can demonstrate that the implementation works on the selected records, but it is not sufficient to estimate how the model will perform on all possible patient conditions. The accuracy result therefore should be presented as a result of the implemented system test rather than as a clinical claim or a guarantee of diagnostic accuracy.

The dataset is another limitation that should be considered. The thesis recommends increasing the amount and variation of training data so that the model can represent a wider range of health conditions. Future evaluation can also compare several K values using a more systematic procedure. The thesis specifically identifies GridSearchCV as a possible approach for automatically searching for an appropriate K value and SMOTE as a possible technique for addressing class imbalance. These approaches are not claimed as part of the current result; they are proposed as directions for subsequent research.

For future implementation, the application can be developed further by adding health notifications, integration with wearable devices, and a mechanism for follow-up consultation. Data protection should also receive greater attention because the application manages health-related information. Stronger authentication, secure storage, regular database backup, and additional access controls can be considered in future versions. These improvements would extend the current application while maintaining the main KNN prediction workflow developed in this research.

Table 4. Relationship between system components and evaluation results

Component	Main role	Evaluation	Result
Authentication	Controls access to the application	Black Box login/logout	Valid
User management	Maintains application user data	Black Box CRUD scenarios	Valid
Training data	Maintains KNN reference records	Black Box CRUD scenarios	Valid
K-value setting	Stores the K parameter used by KNN	Black Box input validation	Valid
Prediction	Processes patient attributes and produces class	Black Box prediction scenarios	Valid
Prediction history	Stores and manages previous results	Black Box history scenarios	Valid
KNN accuracy	Compares prediction with known labels	10 test records	100%

4. CONCLUSION

This research successfully developed a web-based diabetes prediction application that can be used for early screening by entering glucose, blood pressure, BMI, age, and gender parameters. The K-Nearest Neighbor algorithm was successfully integrated into the application. The system calculates Euclidean distance, selects the nearest neighbors according to the configured K value, and produces a diabetes or normal classification. Accuracy testing using 10 test records produced 10 correct predictions and 0 incorrect predictions, resulting in an accuracy of 100%. The result demonstrates that the implemented system performed correctly on the evaluated test records, while further testing with larger and more diverse data is still recommended.

REFERENCES

- [1] A. K. Putra, "Rancang Bangun Sistem Monitoring Efisiensi Pada Mesin Boiler DBU Berbasis Web," *Jurnal Instrumentasi dan Teknologi Informatika (JITI)*, 2021.
- [2] A. Saputra et al., "Tinjauan Sistematis Sistem Informasi Kesehatan Berbasis Web Untuk Transformasi Manajemen Penyakit Kronis," *JTBC: Jurnal Teknologi dan Bisnis Cerdas*, pp. 48–56, 2026.
- [3] E. Effendy et al., "Mengenal Sistem Manajemen Dakwah (Pengertian Sistem, Karakteristik Sistem)," *Jurnal Pendidikan dan Konseling (JPDK)*, vol. 5, no. 2, pp. 4343–4349, 2023.
- [4] F. Ruziq and M. R. Wayahdi, "Web-Based Diabetes Risk Prediction System Using K-NN on Kaggle Early Stage Diabetes Dataset," *JUTIF (Jurnal Informatika)*, pp. 3217–3229, 2025.
- [5] Fathirma'ruf and B. M. Said, "Pengembangan Perangkat Pembelajaran Konstruktivistik Model Teaching with Analogies (TWA) pada Mata Kuliah Database Management System (DBMS) untuk Meningkatkan Kemampuan Berpikir Kreatif Mahasiswa," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 7, no. 5, pp. 1051–1060, 2021.
- [6] Indrawan, 2024.
- [7] J. Parhusip and G. Stevans, "Analisis Perbandingan Akurasi Algoritma K-Nearest Neighbor (KNN), Decision Tree, dan Random Forest dalam Klasifikasi Penyakit Diabetes," *JURNAL MEDIA INFORMATIKA (JUMIN)*, pp. 26–32, 2026.
- [8] Laila, F. and N. A. Hasibuan, "Pemilihan Pengangkatan Karyawan Tetap Menerapkan Metode Multi-Attributive Border Approximation Area Comparison," *Journal of Informatics Management and Information Technology*, pp. 5–12, 2021.
- [9] M. A. A. Candra and I. A. Wulandari, "Sistem Informasi Berprestasi Berbasis Web Pada SMP Negeri 7 Kota Metro," *Jurnal Mahasiswa Ilmu Komputer (JMIK)*, pp. 175–189, 2021.
- [10] Meiyanti, "Rancang Bangun Sistem Informasi Reservasi Kamar Hotel Menggunakan Java NetBeans," *Sisfo: Jurnal Ilmiah Sistem Informasi*, vol. 5, no. 2, 2021.
- [11] M. P. Daulay, R. E. C. Malau, and N. A. Hasibuan, "Sistem Pendukung Keputusan Penentuan Karyawan Terbaik Berdasarkan Kinerja Menggunakan Metode TOPSIS," *TIN: Terapan Informatika Nusantara*, pp. 80–87, 2022.
- [12] Mukhlis, I. Ramadhani, and R. Santoso, "Perancangan Basis Data Perpustakaan Universitas Menggunakan MySQL dengan Physical Data Model dan Entity Relationship Diagram," *Journal of Technology and Informatics (JoTI)*, pp. 81–87, 2023.
- [13] N. Ikhsan, "Klasifikasi Penyakit Diabetes Menggunakan Metode K-Nearest Neighbors (KNN) Berdasarkan Data Klinis," *INFOTECH: Jurnal Informatika & Teknologi*, pp. 19–26, 2026.
- [14] Nuristiqomah, 2021.
- [15] Prahasti, 2022.
- [16] R. Riansyah and A. Syahputra, "Sistem Klasifikasi Diabetes Mellitus Menggunakan Algoritma K-Nearest Neighbor (KNN) Berbasis Web," *JUIKTI (Jurnal Ilmu Komputer dan Teknik Informatika)*, pp. 119–130, 2026.
- [17] Saifudin, 2024.
- [18] Santoso, 2023.
- [19] Setiyani and Lila, "Desain Sistem: Use Case Diagram," *Prosiding Seminar Nasional Inovasi dan Adopsi Teknologi (INOTEK)*, vol. 1, no. 1, 2021.
- [20] S. Narulita, A. Nugroho, and M. Z. Abdillah, "Diagram Unified Modelling Language (UML) untuk Perancangan Sistem Informasi Manajemen Penelitian dan Pengabdian Masyarakat (SIMLITABMAS)," *Jurnal Publikasi Sistem Informasi dan Telekomunikasi*, vol. 2, no. 3, 2024.
- [21] S. Cholarika and F. A. Zahro, "Konsep Dasar Dalam Sistem Informasi Manajemen Dalam Pendidikan," *Al-Bahjah Journal of Islamic Education Management*, pp. 51–60, 2023.
- [22] S. Arfiah, F. Wajid, and N. Nur, "Optimasi Algoritma K-Nearest Neighbors pada Prediksi Penyakit Diabetes," *JURIKOM (Jurnal Riset Komputer)*, pp. 230–240, 2025.

- [23] T. Sutabri and M. T. Sadewa, "Perancangan Sistem Informasi Pendataan Inventory Aset pada Depot Kayu Kusen Laris III," Jurnal Sains dan Teknologi Informasi, vol. 3, no. 1, pp. 11–17, 2025.
- [24] Tjoe, "Metode Waterfall," 2020.
- [25] Y. Rahmawati and M. Syukri, "Faktor Risiko Diabetes Melitus Tipe 2 pada Kelompok Umur ≥ 15 Tahun di Provinsi DKI Jakarta: Analisis Data SKI 2023," Jurnal Penelitian Sains dan Kesehatan Avicenna, pp. 257–267, 2026.