

Article

End-to-End Machine Learning-Based System for Diabetes Monitoring and Prediction Using Mobile Terminals

Alexandra Fanca , Adela Pop *, Alexandru Ciobotaru , Dan Ioan Gota  and Honoriu Valean 

Automation Department, Faculty of Automation and Computer Science, Technical University of Cluj-Napoca, 400114 Cluj-Napoca, Romania; alexandra.fanca@aut.utcluj.ro (A.F.); alexandru.ciobotaru@aut.utcluj.ro (A.C.); dan.gota@aut.utcluj.ro (D.I.G.); honoriu.valean@aut.utcluj.ro (H.V.)

* Correspondence: adela.pop@aut.utcluj.ro

Abstract

Diabetes mellitus is a major chronic disease that requires continuous monitoring and timely risk assessment to reduce the likelihood of severe long-term complications. Recent advances in mobile health technologies and machine learning (ML) provide new opportunities for developing intelligent systems that support diabetes self-management and early risk screening. This paper presents an end-to-end mobile health platform that integrates diabetes monitoring functionalities with an ML-based prediction service within a modular client-server architecture. The proposed system enables users to record glucose measurements, insulin injections, physical activity, and other health-related information while providing historical data visualization, automated reminder notifications, and real-time diabetes risk prediction. The prediction module was trained using the publicly available PIMA Indians Diabetes Dataset and evaluated using Decision Tree (DT), Random Forest (RF), and XGBoost classifiers. Model performance was assessed using accuracy, precision, recall, specificity, F1-score, calibration analysis, and the area under the receiver operating characteristic curve (AUC). Experimental results showed that the RF classifier achieved the highest AUC (0.925), demonstrating superior discrimination capability among the evaluated models and making it the most suitable candidate for deployment within the proposed platform. Although the current prediction model was trained on a benchmark public dataset, the proposed framework provides a practical foundation for integrating ML-driven decision support into mobile health applications and can be further extended through external clinical validation and personalized prediction models.

Keywords: diabetes monitoring; mobile health; diabetes risk prediction; machine learning; Random Forest; self-management



Academic Editor: Alexander Zhbanov

Received: 18 June 2026

Revised: 7 July 2026

Accepted: 10 July 2026

Published: 12 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Chronic diseases represent a major global public health challenge and account for approximately 90% of deaths worldwide. Among these conditions, diabetes mellitus is one of the most prevalent metabolic disorders, affecting millions of people globally. Diabetes occurs when the body is unable to adequately regulate blood glucose levels owing to insufficient insulin production, impaired insulin action, or both. Without appropriate treatment and continuous monitoring, diabetes may lead to severe long-term complications, including cardiovascular disease, chronic kidney disease, neuropathy, and retinopathy [1].

Effective diabetes management relies on continuous monitoring of blood glucose levels together with careful observation of lifestyle-related factors. Regular monitoring

enables patients and healthcare professionals to make informed decisions regarding medication, diet, and physical activity. Previous studies have demonstrated that regular glucose monitoring significantly reduces the risk of long-term diabetes-related complications [2]. Self-monitoring of blood glucose is considered one of the most effective methods for managing diabetes. Without continuous monitoring and appropriate treatment, the disease can significantly impact daily activities and overall quality of life. Current monitoring techniques include traditional self-monitoring of blood glucose, continuous glucose monitoring systems, and telehealth solutions that enable remote supervision by healthcare professionals [2–4]. In recent years, mobile health technologies have emerged as powerful tools for supporting patients in managing chronic diseases. Mobile health applications enable users to record health-related data, receive personalized feedback, and visualize longitudinal trends in their medical information, thereby supporting continuous disease monitoring, improving patient engagement, and increasing adherence to treatment plans. Concurrently, machine learning (ML) techniques have demonstrated considerable potential for supporting medical decision-making and disease risk prediction [5,6]. ML algorithms are capable of identifying complex relationships among clinical and demographic variables associated with diabetes risk. Numerous studies have applied algorithms such as Support Vector Machines, DTs, and Neural Networks for diabetes prediction, achieving promising results [7–10].

The integration of predictive models within mobile health platforms can significantly improve early detection and patient awareness. Motivated by these developments, this paper proposes a mobile-based diabetes monitoring and prediction system that combines ML algorithms with mobile health technologies. The proposed platform is primarily intended to support diabetes self-management while providing machine learning-based diabetes risk assessment as a decision-support functionality. The prediction module is designed to assist users in identifying potential diabetes risk and does not replace clinical diagnosis or medical decision-making. For individuals already diagnosed with diabetes, the application facilitates continuous monitoring through glucose logging, insulin tracking, physical activity recording, and automated reminder notifications. Although considerable progress has been achieved in both mobile health technologies and machine learning-based diabetes prediction, most published studies focus either on predictive model development or on mobile monitoring applications as separate solutions. Comparatively fewer studies integrate these components into a unified end-to-end platform capable of supporting continuous diabetes self-management while simultaneously providing real-time diabetes risk prediction.

Several recent studies have investigated the application of ML models for diabetes prediction using clinical and lifestyle data. Comparative analyses have shown that ensemble models and deep learning architectures often achieve high accuracy in predicting the onset of type 2 diabetes [11,12].

In addition to predictive capabilities, modern digital health systems increasingly focus on continuous monitoring and self-management support for chronic diseases. Mobile technologies provide patients with accessible tools for tracking daily health parameters such as glucose levels, insulin administration, and physical activity. These systems can significantly improve disease awareness and allow patients to maintain a consistent history of medical data that can later be used during clinical consultations. Mobile applications are particularly suitable for diabetes management since patients typically carry smartphones throughout the day, making them an effective platform for real-time monitoring and feedback [13,14].

Despite these advances, relatively few studies integrate machine learning-based diabetes risk prediction with comprehensive mobile health functionalities within a single

end-to-end platform. Developing such integrated solutions has the potential to improve both early risk assessment and long-term disease self-management by combining predictive analytics with continuous health monitoring and user engagement. The main contributions of this work are as follows:

- Design and implementation of an integrated mobile health system for diabetes monitoring and prediction.
- Design of a scalable client-server architecture combining mobile technologies with ML services.
- Development of a REST-based prediction service that enables real-time prediction and risk assessment of diabetes.
- Integration of comprehensive diabetes self-management functionalities including glucose logging, insulin tracking, and physical activity monitoring.
- Design and implementation of a reminder mechanism that encourages continuous glucose monitoring and improves patient adherence.

The remainder of this paper is organized as follows. Section 2 presents related work in diabetes monitoring and ML-based prediction models. Section 3 describes the materials and methods used in the development of the predictive model. Section 4 presents the architecture of the proposed system. Section 5 presents the experimental evaluation of the proposed system, while Section 6 discusses the obtained results, limitations, and future research directions. Finally, Section 7 concludes the paper by summarizing the main findings and outlining potential directions for future work.

2. Related Work

2.1. Diabetes Monitoring Systems

Monitoring blood glucose levels is essential for effective diabetes management. Patients diagnosed with diabetes are required to regularly track their glucose levels in order to maintain optimal glycemic control and prevent complications. Previous studies have shown that individuals who experience diabetes-related complications tend to monitor their glucose levels more frequently due to increased awareness regarding the importance of disease management [15].

Certain patient groups require even more frequent monitoring. For example, pregnant women diagnosed with gestational diabetes must closely monitor their glucose levels to prevent complications that may affect both the mother and the fetus. Medical guidelines recommend regular blood glucose evaluations throughout pregnancy to ensure appropriate treatment and lifestyle adjustments [16].

In recent years, mobile health applications have become increasingly popular tools for diabetes monitoring. These applications enable users to record various health parameters, including glucose levels, medication intake, physical activity, and dietary habits. Furthermore, the integration of mobile applications with wearable sensors and Internet of Things (IoT) devices allows continuous monitoring and real-time analysis of physiological data.

Research has shown that combining continuous glucose monitoring systems with mobile applications can improve glycemic control and reduce the risk of severe hypoglycemic events. In addition, elderly patients using IoT-enabled monitoring solutions have demonstrated improved disease management compared to those relying on traditional monitoring methods [17].

Systematic reviews have also highlighted the benefits of mobile health systems in diabetes care, emphasizing their role in improving patient education and providing timely feedback for self-management. These technologies contribute to better adher-

ence to treatment plans and improved long-term health outcomes for patients with diabetes [13,14].

Although prior studies have reported strong predictive performance for diabetes risk assessment, many focus either on standalone prediction models or on monitoring applications without integrated predictive analytics. In contrast, the proposed system combines mobile health monitoring, patient data logging, visualization, reminders, and ML-based prediction within a unified client-server architecture.

2.2. Machine Learning for Diabetes Prediction

A meta-analysis conducted by Kodama et al. demonstrated the strong predictive capabilities of modern ML algorithms compared with traditional statistical approaches for diabetes risk assessment [8].

For example, Farran et al. developed ML models capable of predicting the risk of diabetes and hypertension using national healthcare datasets [7]. Similarly, Qin et al. showed that ML algorithms trained on lifestyle-related variables can effectively predict diabetes risk [9].

Additional studies have shown that incorporating lifestyle variables into predictive models significantly improves the accuracy of diabetes prediction systems. Health-related behaviors such as diet, physical activity, and body weight represent important predictors of diabetes development and can enhance the performance of ML models when included as predictive features [10].

Hybrid ML approaches have also been explored to further improve prediction performance. For instance, Mastoli et al. proposed a hybrid model combining Support Vector Machines and Artificial Neural Networks, achieving high prediction accuracy and demonstrating the effectiveness of combining multiple algorithms for diabetes diagnosis tasks [18].

Other research has investigated the use of electronic health records to develop predictive models capable of identifying patients at risk of missing clinical appointments or experiencing complications related to diabetes. Kurasawa et al. showed that ML techniques applied to clinical records can assist healthcare systems in identifying patients who require additional monitoring or medical intervention [19].

Recent studies have investigated the use of advanced ML and deep learning techniques for diabetes prediction, demonstrating high predictive accuracy and improved interpretability of models [5,6,11,12,20,21].

Despite the significant progress achieved in diabetes prediction using ML algorithms, many existing studies focus primarily on the development of predictive models without integrating them into practical healthcare systems. Moreover, limited research has explored the combination of ML prediction models with mobile health applications designed for continuous patient monitoring and self-management. Therefore, integrating predictive analytics with mobile health platforms represents an important research direction for improving diabetes monitoring and enabling earlier disease detection.

3. Materials and Methods

3.1. Dataset

The ML model used in this study was trained using the PIMA Indians Diabetes Dataset, which is publicly available through the Kaggle platform [22]. This dataset is widely used in diabetes prediction research and contains 768 patient records, together with several clinical attributes commonly used for diabetes risk assessment. The PIMA Indians Diabetes Dataset was selected because it is one of the most extensively used public benchmark datasets

for diabetes prediction, enabling direct comparison with previously published machine learning studies.

The data set includes the following independent variables: number of pregnancies, plasma glucose concentration, blood pressure (mm Hg), skin thickness (mm), insulin level ($\mu\text{U/mL}$), body mass index, diabetes pedigree function, and age. Within the mobile application, BMI is not entered directly by the user but is automatically computed from the user's weight and height before the prediction request is submitted. The target variable indicates whether the patient has diabetes or not. Therefore, diabetes risk prediction can be modeled as a binary classification problem. It should be noted that the health data recorded through the mobile application are currently stored exclusively for monitoring and visualization purposes and are not used to retrain or personalize the deployed machine learning model. Future work will investigate personalization strategies based on longitudinal patient data collected through the application.

To provide a better overview of the features in the data set, Figure 1 shows the correlation matrix between the dependent and independent features. It can be seen that the features that are mostly correlated with the outcome are represented by the glucose level, BMI, and age. From a clinical perspective, these features directly influence the predisposition of a person to diabetes. For example, a high BMI value associated with a high glucose level would significantly increase the chances of developing diabetes. Similarly, a high glucose level is associated with a higher chance of developing diabetes from a clinical perspective.

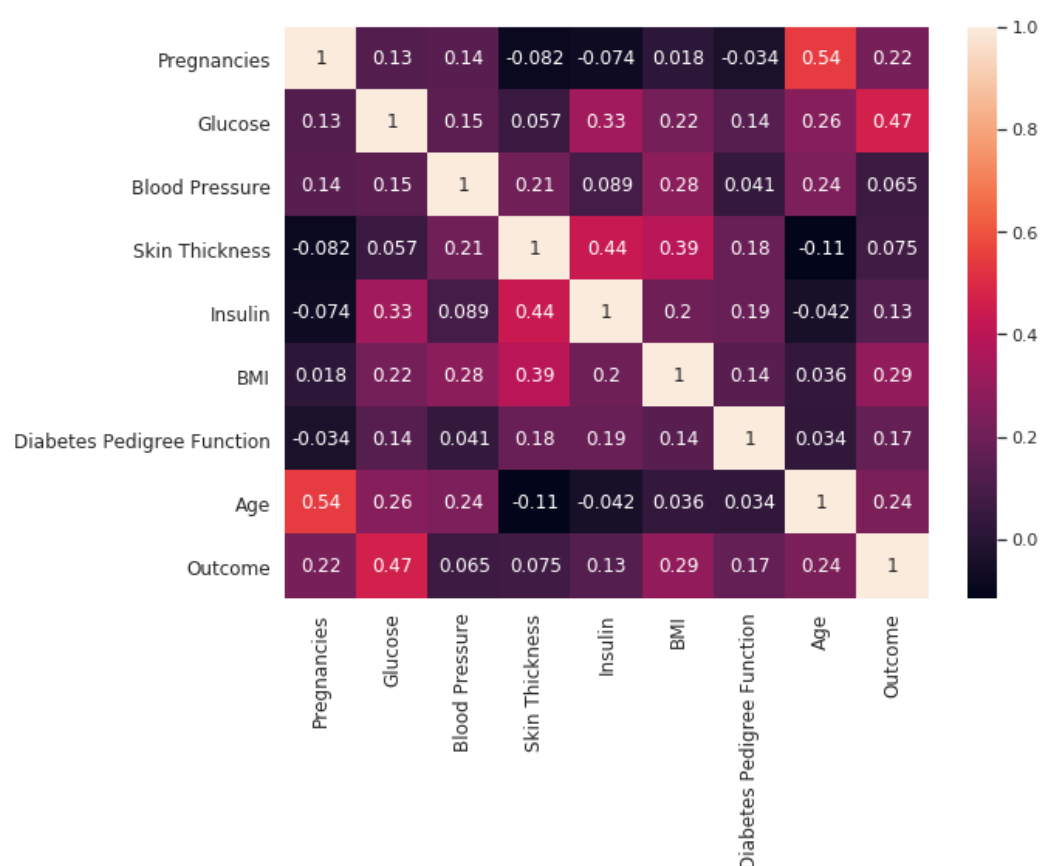


Figure 1. Correlation Matrix of the features provided by the Pima Indians Diabetes Dataset.

3.2. Data Preprocessing

Before training the ML models, the dataset was analyzed to detect inconsistencies, missing values, and potential outliers. Data preprocessing is a critical stage in ML pipelines

because it improves data quality and allows the model to better identify meaningful patterns in the dataset.

The overall data preprocessing and prediction workflow used in this study is illustrated in Figure 2.

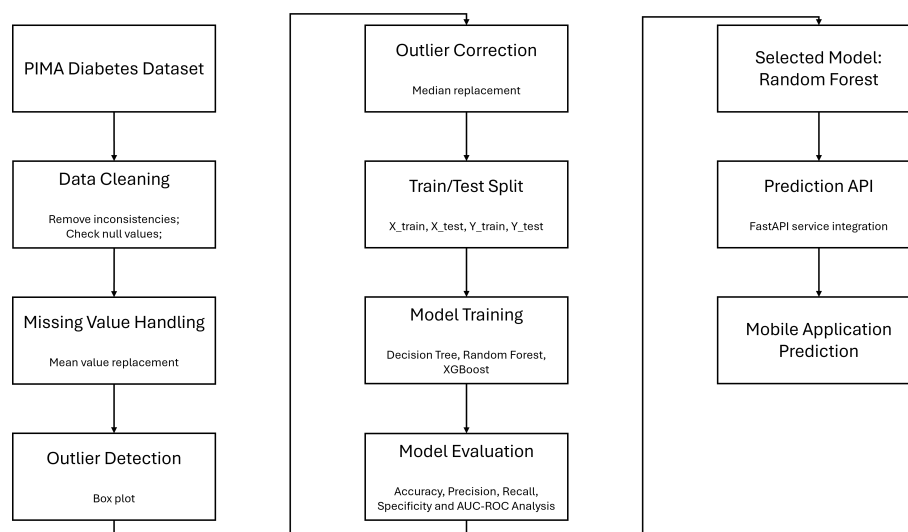


Figure 2. Data preprocessing and prediction workflow used for the diabetes prediction.

The workflow begins with the PIMA dataset, followed by preprocessing steps including missing value handling and outlier correction. After preprocessing, the dataset is divided into training and testing subsets. Multiple ML algorithms are then trained and evaluated in order to determine the most suitable model for diabetes prediction.

During the exploratory analysis, several attributes were found to contain zero values. Since zero values are physiologically impossible for certain medical measurements, such as glucose concentration or insulin level, these values were interpreted as missing data. The distribution of zero values across the dataset attributes is illustrated in Figure 3.

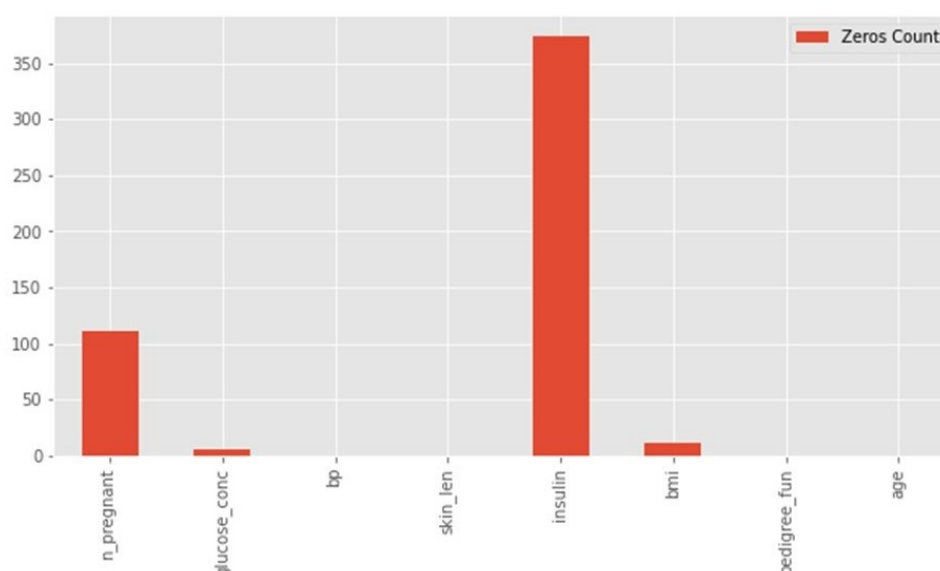


Figure 3. Distribution of zero values across the clinical variables in the PIMA Indians Diabetes Dataset.

The analysis revealed the presence of extreme values for several attributes, including blood pressure, BMI, skin thickness, and age. To reduce the influence of these extreme

observations on the training process, the detected outliers were replaced with median values calculated for each attribute. After completing these preprocessing steps, the dataset was considered suitable for training ML models.

3.3. Machine Learning Algorithms

Three supervised ML algorithms were evaluated in order to determine the most suitable model for diabetes prediction: Decision Tree (DT), Random Forest (RF), and XGBoost. DT is a supervised learning algorithm that predicts a target value by learning decision rules derived from the dataset features. RF is an ensemble learning method that constructs multiple DTs and combines their outputs to improve prediction accuracy. XGBoost is a gradient boosting algorithm that builds sequential DTs and optimizes prediction performance using gradient descent techniques.

4. Proposed System Architecture

The overall workflow of the proposed diabetes monitoring and prediction system is illustrated in Figure 4.

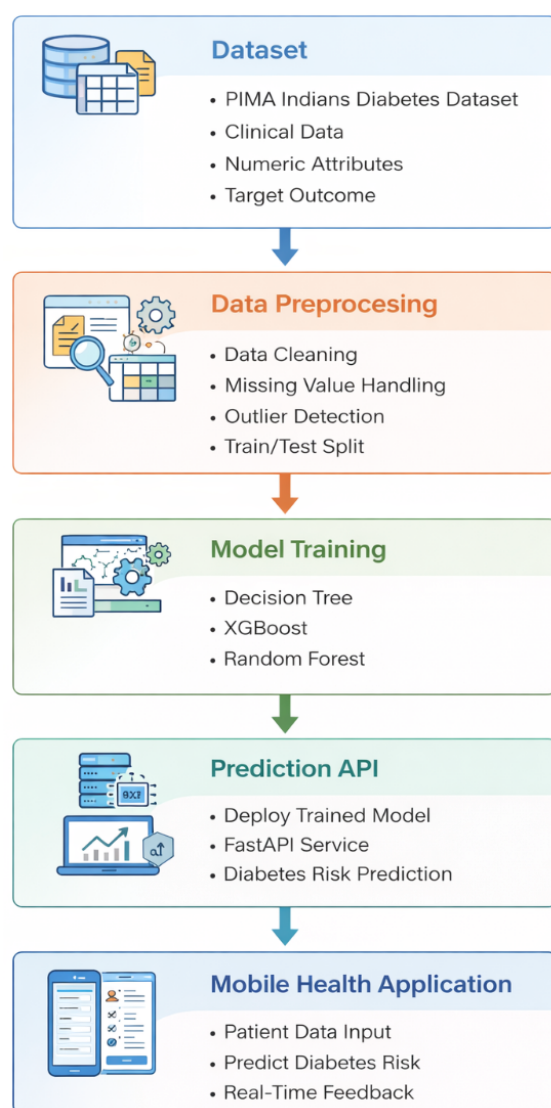


Figure 4. Machine learning workflow architecture of the proposed system.

The workflow begins with the PIMA Indians Diabetes Dataset, followed by preprocessing steps including data cleaning, missing value handling, and outlier detection. The

processed dataset is then used to train multiple ML algorithms. The best-performing model is deployed through a FastAPI-based prediction service and integrated into the mobile health application to provide real-time diabetes risk prediction.

The proposed system follows a modular client-server architecture designed to support both health monitoring and predictive analytics functionalities. The architecture consists of three main components: a mobile client application, a backend application server, and a ML prediction service. The mobile application allows users to record health-related information such as glucose levels, insulin injections, and physical activity. In addition, users can request diabetes risk predictions based on medical parameters entered into the application. The backend server is responsible for managing authentication, data storage, and communication between the mobile application and the prediction service. The prediction service hosts the trained ML model and processes prediction requests sent by the backend server. The overall system architecture is illustrated in Figure 5.

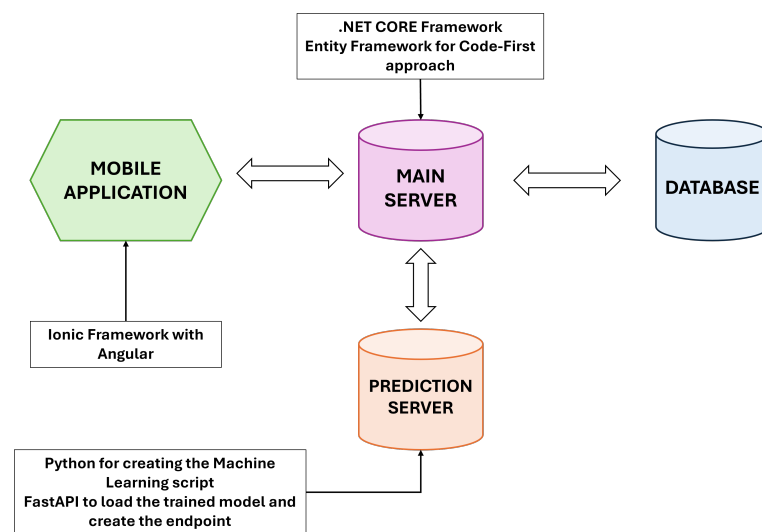


Figure 5. Overall architecture of the proposed diabetes monitoring and prediction system.

The mobile client communicates with the backend server implemented using ASP.NET Core through RESTful APIs. The backend server interacts with the ML prediction service implemented using Python version 3.14.6 and FastAPI, version 0.139.0 to generate diabetes risk predictions.

4.1. System Objectives and Functionalities

The main objective of the proposed system is to support diabetes monitoring and prediction through the integration of mobile health technologies and ML algorithms.

The system allows users to obtain a diabetes risk prediction by completing a form containing relevant clinical and demographic parameters required by the trained ML model. Based on these inputs, the system generates a prediction regarding the likelihood of diabetes occurrence. The prediction form collects the user's weight and height together with the remaining clinical parameters required by the deployed machine learning model. Since the prediction model was trained using the Body Mass Index (BMI) feature provided by the PIMA Indians Diabetes Dataset, the application automatically computes BMI from the entered weight and height values using the standard Body Mass Index formula. The calculated BMI value is then transmitted to the prediction service together with the remaining input variables, ensuring full consistency between the mobile application and the machine learning model.

In addition to prediction functionality, the application enables continuous monitoring of health-related data. Users can log glucose levels, insulin injections when required, and

physical activity. These records are stored in the system and can later be accessed through historical logs and graphical visualizations that help users observe trends and fluctuations in their health data.

The application also includes a calendar feature that allows users to schedule future medical appointments or health-related events. Another important functionality is the automated notification mechanism, which sends reminders if users do not record glucose data by a predefined time during the day. This feature encourages consistent monitoring and supports better diabetes self-management.

4.2. Use Case Overview

The primary features of the application include logging glucose levels and insulin injections, as well as requesting predictions based on medical parameters entered by the user. The prediction functionality allows users to analyze their potential diabetes risk and increase awareness regarding their health status.

The interactions between the user and the system functionalities are illustrated in the use case diagram presented in Figure 6.

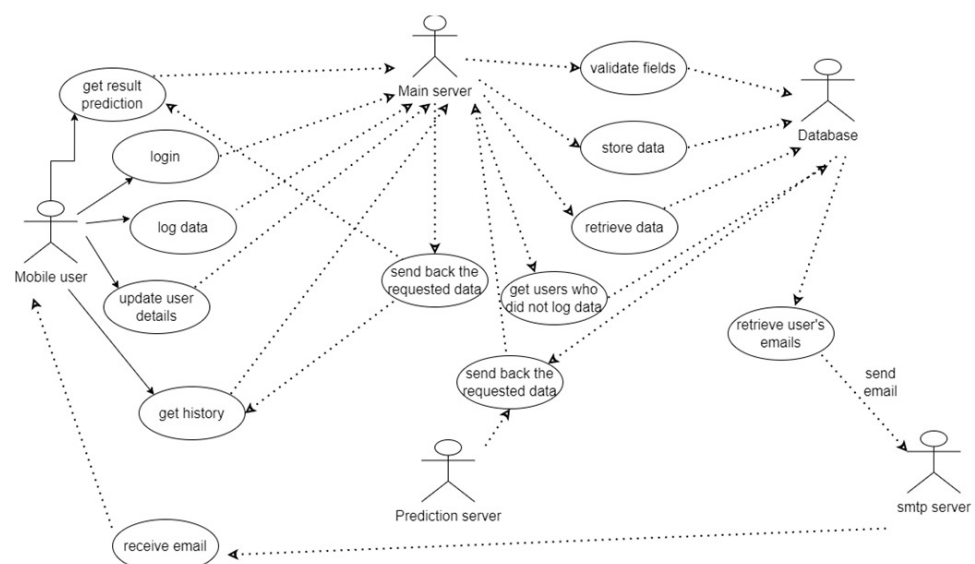


Figure 6. Use case diagram of the proposed diabetes monitoring system.

Additional functional requirements include validation mechanisms implemented both on the client side and on the server side to ensure data integrity. The application also stores all user-related data and monitoring logs in the system database. The application allows users to update their personal information and ensures that restricted routes remain accessible only to authenticated users. If an unauthenticated user attempts to access a protected route, the system automatically redirects them to the login page.

4.3. Implementation Technologies

The proposed system was implemented using a combination of modern web and mobile development technologies. The mobile client was developed using the Angular-integrated Ionic framework, enabling cross-platform deployment on Android and iOS devices. Ionic provides a rich set of UI components and allows the development of hybrid mobile applications using web technologies such as HTML, CSS, and TypeScript.

The backend server responsible for data management and user authentication was implemented using the ASP.NET Core framework. This framework enables the development of scalable RESTful APIs and supports modern software architecture patterns such as dependency injection and repository-based data access.

The ML prediction service was implemented using Python and the FastAPI framework. FastAPI allows efficient deployment of trained ML models as web services and provides automatic API documentation and testing capabilities.

For data persistence, the system uses a Microsoft SQL Server database accessed through Entity Framework using the code–first approach. This approach simplifies database management and ensures consistency between application models and database schema.

4.4. Database Design

The system database consists of six main tables that store information related to users and their health monitoring data. The Users table stores general user information, including name, date of birth, username, password, and diabetes–related information. The Predictions table stores the prediction results generated for each user. The GlucoseLogs, InsulinLogs, and SportLogs tables store monitoring data related to glucose levels, insulin intake, and physical activity. Additionally, the Events table stores information about scheduled medical appointments or other health–related events. All tables are connected to the Users table, which acts as the central entity of the database schema. The structure of the database is illustrated in Figure 7.

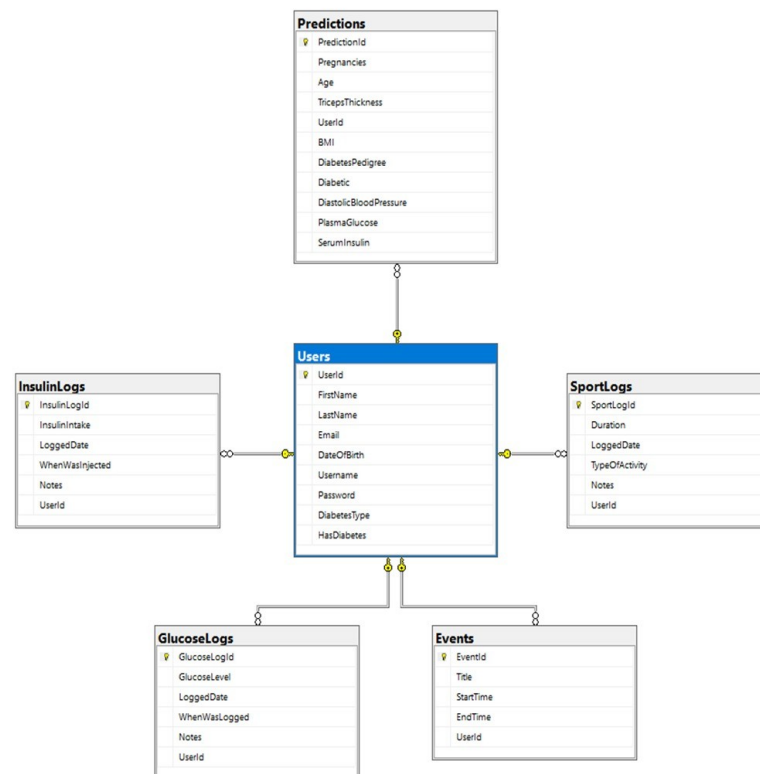


Figure 7. Database schema used for storing user health data and monitoring logs.

4.5. System Workflow

The interaction between the mobile application, backend server, and prediction service is illustrated in the sequence diagram shown in Figure 8. The backend server was implemented using the ASP.NET Core framework following the Model–View–Controller (MVC) architectural pattern. This approach improves modularity and maintainability by separating the application logic into distinct components responsible for data handling, request processing, and response generation. To manage database operations, the system employs the repository pattern, which centralizes data access logic and reduces code duplication across different components. The backend communicates with the mobile application through RESTful APIs and processes requests encoded in JSON format.

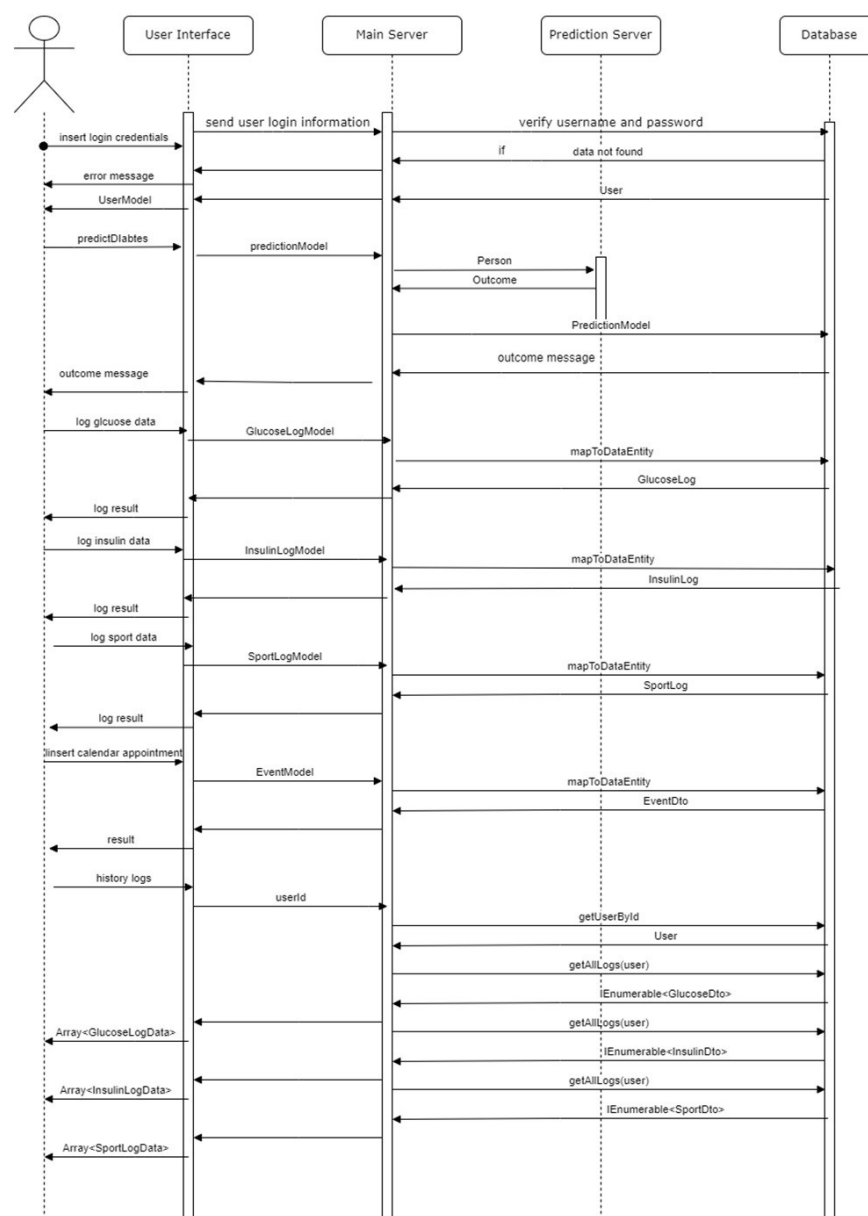


Figure 8. Sequence diagram illustrating the interaction between the mobile client, backend server, and prediction service.

This workflow describes the main operational process of the system. The user interacts with the mobile application to submit monitoring data or request predictions. The backend server processes the request and communicates with the prediction service when a prediction is required. The prediction service evaluates the input parameters using the trained ML model and returns the result to the server, which forwards the response to the mobile application.

4.6. Notifications Mechanism

In addition to the core functionalities of monitoring and prediction, the system includes an automated notification mechanism. The system sends daily reminder emails to users who have not recorded glucose data by a specified time during the day. Maintaining a consistent history of glucose measurements is essential for effective diabetes management, as it allows users to identify trends and variations in their glucose levels over time. An example of the notification email sent by the system is presented in Figure 9.

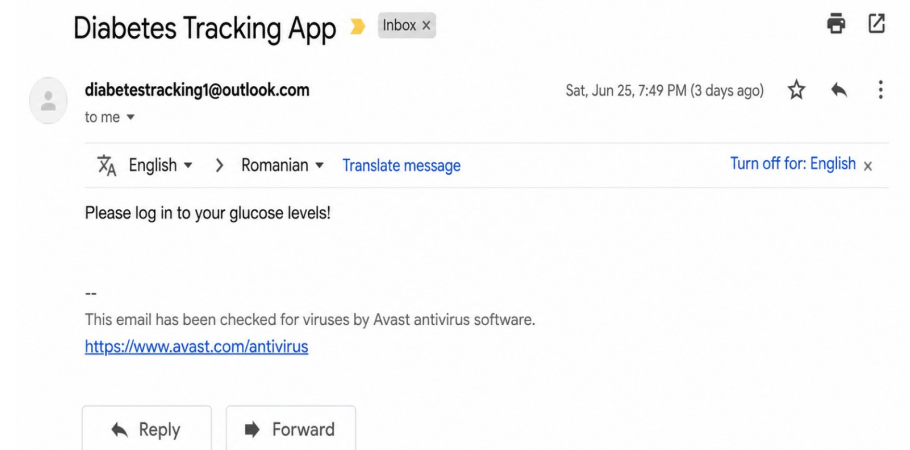


Figure 9. Example of an automated reminder notification email.

The notification service is implemented using a scheduled job system that periodically checks whether users have logged glucose values during the day. If no data have been recorded before a predefined time, the system automatically sends a reminder email encouraging the user to perform the measurement. This mechanism promotes consistent monitoring behavior, which is essential for effective diabetes self-management.

5. Experimental Results

5.1. Prediction Performance

In order to deploy the prediction service within the proposed system, an ML model capable of accurately predicting diabetes risk was required. The model was trained using the PIMA Indians Diabetes Dataset, which contains 768 patient records and several clinical attributes commonly used for diabetes prediction [22]. Following the preprocessing phase described in Section 3, the dataset was divided into training and testing subsets. The feature matrices were stored in X_{train} and X_{test} , while the corresponding target vectors were stored in y_{train} and y_{test} . Three supervised ML algorithms were evaluated in order to determine the most suitable model for diabetes prediction: RF, DT, and XGBoost. Each algorithm was trained using the training dataset and evaluated using the testing dataset. Additionally, to increase the reliability of the evaluation of the ML models, 5-fold cross-validation was employed. The performance of the evaluated classifiers was assessed using six standard evaluation metrics: Accuracy (ACC), Positive Predictive Value (PPV, also referred to as Precision), Recall (REC, also referred to as the True Positive Rate (TPR) or Sensitivity), Specificity (SPC, also referred to as the True Negative Rate (TNR)), F1-score, and the Area Under the Receiver Operating Characteristic Curve (AUC-ROC).

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

$$PPV = \frac{TP}{TP + FP} \quad (2)$$

$$TPR = REC = \frac{TP}{TP + FN} \quad (3)$$

$$TNR = SPC = \frac{TN}{TN + FP} \quad (4)$$

$$F1 = 2 \cdot \frac{PPV \cdot TPR}{PPV + TPR} \quad (5)$$

$$\text{AUC-ROC} = \int_0^1 \text{TPR}(\text{FPR}) d(\text{FPR}) \quad (6)$$

where TP, TN, FP, and FN denote the numbers of true positives, true negatives, false positives, and false negatives, respectively. The False Positive Rate (FPR) is defined as:

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}} \quad (7)$$

The Receiver Operating Characteristic (ROC) curve represents the relationship between the True Positive Rate (TPR) and the False Positive Rate (FPR) across different classification thresholds.

The results obtained by the DT, RF, and XGBoost models are presented in Table 1. Moreover, the corresponding confusion matrices and ROC curves are presented in Figures 10 and 11, respectively. The Area Under the ROC Curve (AUC–ROC) provides a quantitative measure of the classifier’s overall discriminative performance, with higher values indicating better classification capability.

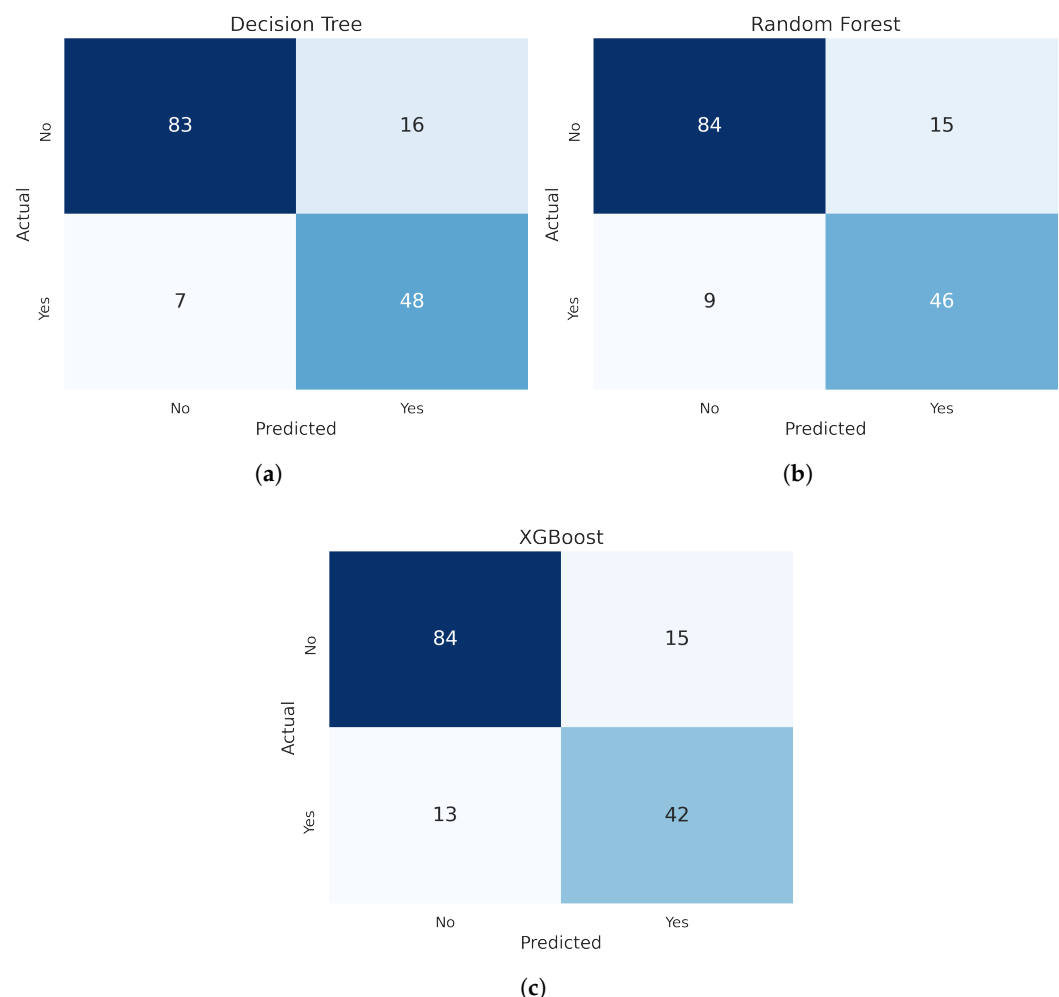


Figure 10. Confusion Matrices of the: (a) Decision Tree, (b) Random Forest and (c) XGBoost Classifiers.

Additionally, the calibration curves presented in Figure 12 compare the probabilities predicted by the ML models with the observed outcome frequencies. Five bins were used to generate the calibration curves. For each bin, two quantities were employed to determine the calibration level of the ML–based models. The first quantity is represented by the mean predicted probability which is measured on the x–axis and highlights the average predicted probability assigned by each ML model. The second quantity is represented

by the fraction of positives which is measured along the y-axis and shows the observed proportion of positive-class samples within a certain bin. The calibration curves indicate that the DT and RF models exhibit better agreement between predicted probabilities and observed outcomes than the XGBoost model, which is consistent with the performance metrics reported in Table 1.

Table 1. Performance evaluation results of the machine learning models.

Model	ACC [%]	PPV [%]	TPR/REC [%]	TNR/SPC [%]	F1 [%]	AUC-ROC
Decision Tree	87.3	86.1	87.5	81.6	86.8	0.919
Random Forest	84.5	82.7	83.1	86.9	82.9	0.925
XGBoost	82.2	80.4	81.6	84.8	80.9	0.923

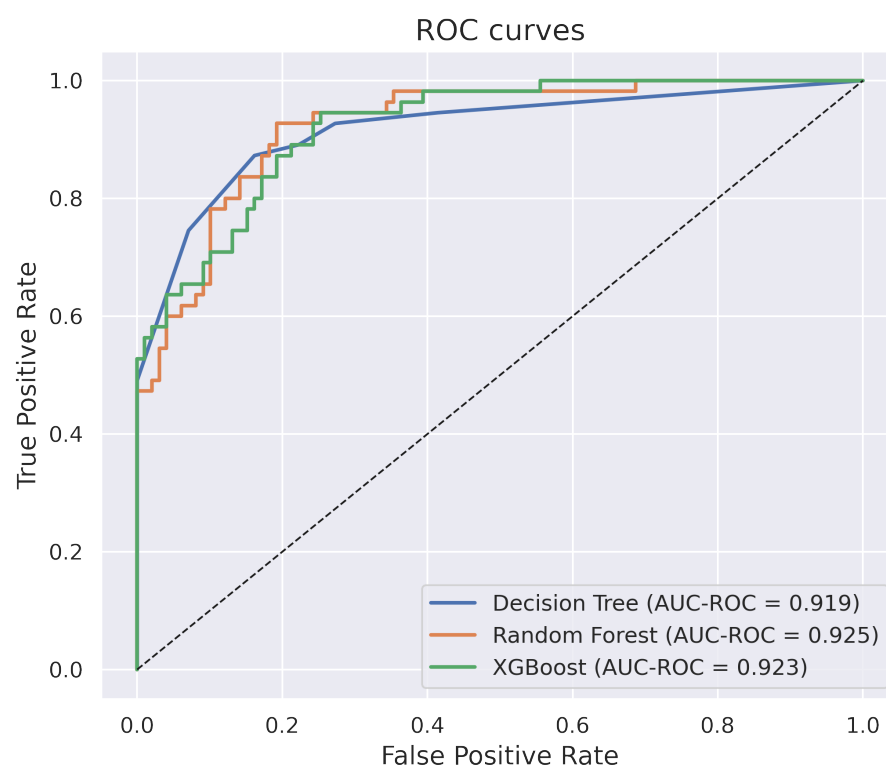


Figure 11. ROC Curve obtained after training the machine learning models.

Although the Decision Tree achieved the highest accuracy, the Random Forest classifier obtained the highest AUC–ROC value, which indicates better discrimination capability across different classification thresholds. Therefore, Random Forest was selected for deployment.

A perfectly calibrated model follows the diagonal line, indicating that predicted probabilities closely match the observed outcomes.

To improve the interpretability of the deployed prediction model, permutation feature importance was computed to identify the variables with the greatest influence on the prediction results. Therefore, Figure 13 presents the top five most important features that are used in the decision-making process using permutation importance. Permutation importance quantifies the importance of each feature's contribution in the dataset by measuring the decrease in the AUC–ROC value as its corresponding values are randomly shuffled.

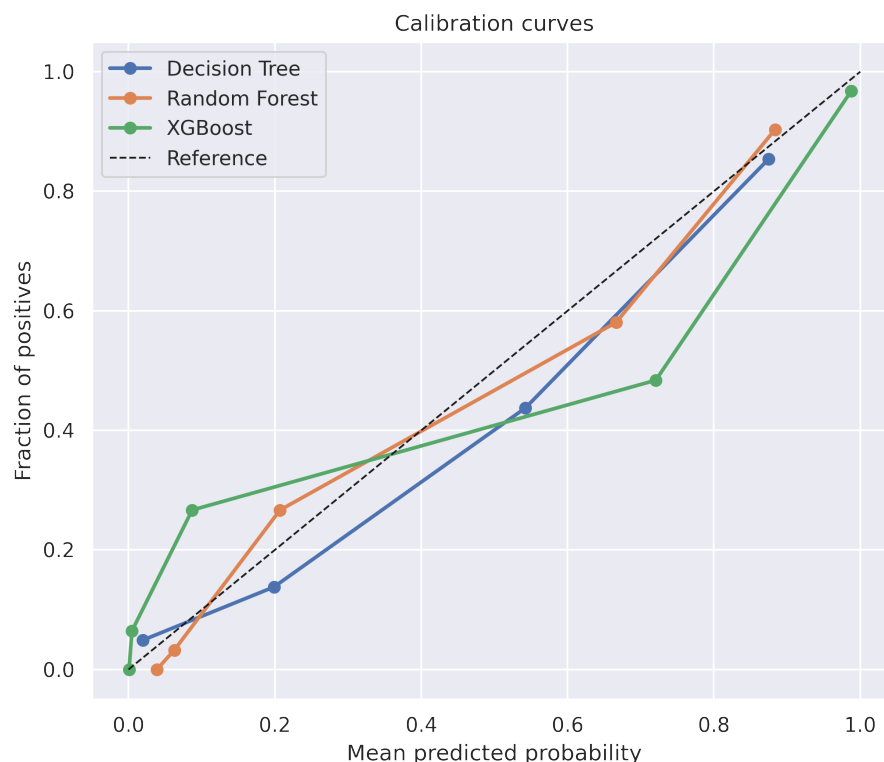


Figure 12. Calibration curves obtained for the evaluated machine learning models.

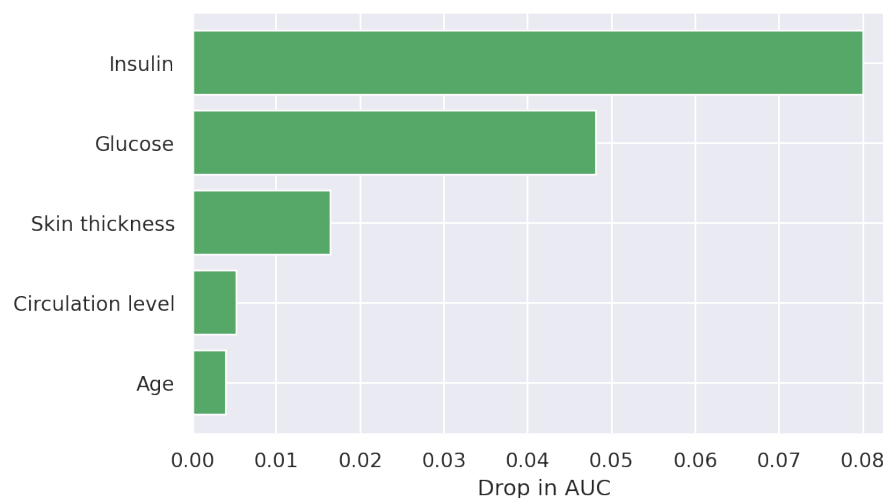


Figure 13. Permutation importance analysis of the five most influential features.

Therefore, the larger the drop in AUC–ROC score, the greater the importance of the feature is. It can be observed that the first two features (e.g., insulin and glucose) are the most important features with respect to the AUC–ROC values. From the clinical point of view, the feature importance is valid since both the glucose and insulin levels directly influence whether a person has diabetes or not.

5.1.1. Functional Validation of the Mobile Application

In addition to evaluating the predictive model, functional testing of the mobile application (Figure 14) was performed in order to validate the correct operation of the implemented features.



Figure 14. User interfaces illustrating (a) the diabetes prediction form, (b) the glucose logging page, and (c) the confirmation message displayed after recording a glucose measurement.

Functional testing was conducted to validate the main components of the application, including user registration, authentication, prediction requests, and health data logging. Validation mechanisms were implemented both on the client side and on the server side to ensure that required fields were completed and that data formats were correct. The ML model was also tested using multiple input scenarios to verify the correctness of the prediction service and the communication between the mobile application and the backend system.

User registration was tested through a form-based interface requiring users to enter their personal information. Validation mechanisms ensure that all required fields are completed, the email address follows a valid format, and the password meets minimum security requirements. Security mechanisms were implemented to protect user data stored within the system. User credentials are encrypted before being stored in the database, reducing the risk of unauthorized access in the event of security breaches. Access control is enforced through authentication mechanisms that restrict protected routes to authenticated users only. In addition, server-side validation is applied to ensure data integrity and prevent invalid or malicious requests.

The backend server verifies whether the username and email address are unique before creating a new user account. User authentication was also validated. When login credentials are submitted, the server checks whether the provided username exists in the database and verifies the corresponding password. If authentication is successful, the user is redirected to the application home page; otherwise, an error message is displayed.

Additional validation mechanisms are implemented for prediction requests and health data logging. For example, prediction requests require all input fields to be completed before submission. Similarly, forms used for logging glucose levels, insulin injections, and physical activity only enable submission once all mandatory fields are filled.

The mobile application also provides a history page that displays glucose measurements, insulin injections, and physical activity records collected during the previous two months. These historical records enable users to monitor changes in their health status over time and may assist healthcare professionals during follow-up consultations (Figure 15). Additionally, the application includes a charts page that graphically presents trends in

blood glucose levels, insulin administration, and physical activity over the same period, allowing users to better understand the evolution of their recorded health parameters.

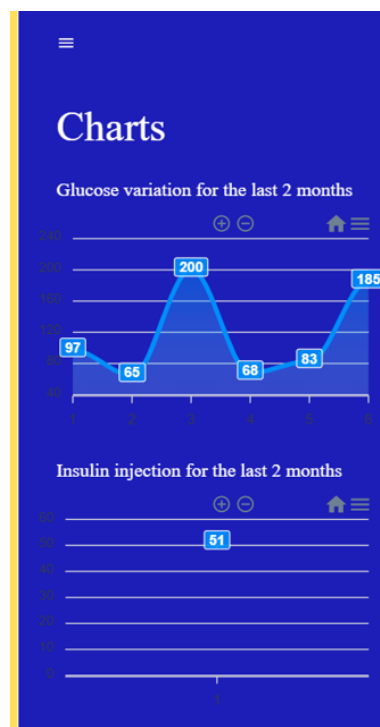


Figure 15. History page displaying the user's blood glucose measurements, insulin injections, and physical activity records collected during the previous two months.

5.1.2. Data Preprocessing Validation

During the preprocessing phase, the dataset was analyzed for potential outliers using box plot visualization for each feature. In some attributes, such as blood pressure, outliers were identified. After applying the outlier correction strategy described in Section 3, the box plot analysis was repeated to verify that the issue had been resolved. An example of the outlier correction process is illustrated in Figure 16.

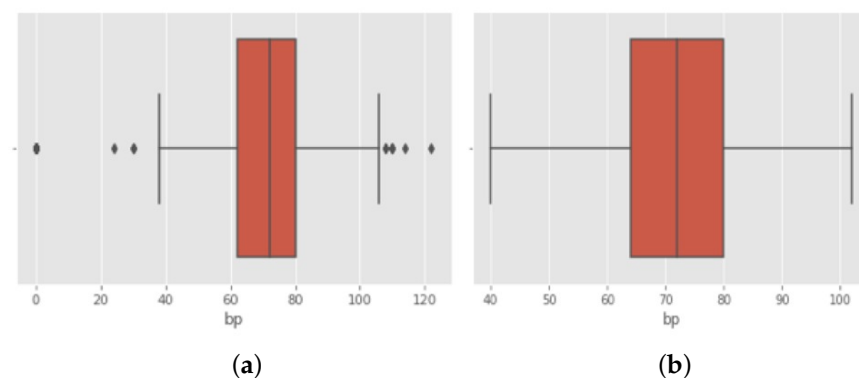


Figure 16. Overview of outlier correction process during data preprocessing: (a) before and (b) after outlier correction.

The corrected dataset was then used for training and evaluating the ML models. Overall, the predictive model achieved high accuracy on the testing dataset, with the RF classifier demonstrating the best performance. In addition, functional testing confirmed that the mobile application correctly supports user registration, authentication, prediction requests, and health data logging.

6. Discussion

The experimental results demonstrate that the RF algorithm achieved the highest AUC score among the evaluated models, outperforming both the DT and XGBoost classifiers. This result highlights the effectiveness of ensemble-based learning techniques in handling complex relationships within medical datasets. RF combines the predictions of multiple DTs, reducing overfitting and improving model stability, which explains its superior performance on the PIMA Indians Diabetes Dataset. These findings are consistent with previous studies reporting strong predictive performance of ML-based diabetes prediction models [20,21].

Beyond the predictive performance of the ML model, an important contribution of the proposed work lies in the integration of the predictive model within a mobile health monitoring platform. While many existing studies focus primarily on the development and evaluation of ML models, the proposed system extends this approach by embedding predictive analytics into a practical mobile health application designed for continuous patient monitoring. This integration enables users not only to obtain predictions regarding their potential diabetes risk but also to monitor relevant health parameters over time, thus supporting proactive disease management.

The system supports diabetes self-management by providing functionalities that allow users to record glucose measurements, insulin injections, and physical activity. These monitoring features are complemented by data visualization tools that enable users to observe trends and variations in their glucose levels through graphical representations and historical logs. Such capabilities can improve patient awareness and facilitate more informed health-related decisions.

Another important functionality of the system is the automated notification mechanism. By sending reminder emails when glucose measurements are not recorded during the day, the system encourages users to maintain consistent monitoring behavior. Continuous monitoring is widely recognized as a critical component of effective diabetes management, as it enables early detection of abnormal glucose levels and helps patients maintain better glycemic control.

Despite the promising results obtained in this study, several limitations should be considered. The predictive model was trained using the PIMA Indians Diabetes Dataset, which contains a relatively limited number of samples and represents a specific population group. Consequently, although the obtained results are encouraging, the generalizability of the model to ethnically and clinically diverse populations has not yet been demonstrated. Future work will therefore focus on external validation using publicly available multi-ethnic datasets, such as NHANES, together with retrospective and prospective validation using real clinical data collected in collaboration with healthcare institutions. Although multiple evaluation metrics were considered, including accuracy, precision, recall, specificity, F1-score, calibration analysis, confusion matrices, and AUC, further validation using independent clinical datasets remains necessary before the model can be considered suitable for deployment in real clinical environments.

From a clinical perspective, the proposed system demonstrates the potential of integrating ML prediction models with mobile health technologies to support early risk detection and continuous monitoring of diabetes. By providing users with a convenient mobile platform for logging glucose levels, insulin injections, and physical activity, the system encourages regular self-monitoring and increases patient awareness regarding their health status. The integration of predictive analytics allows users to receive an estimation of their potential diabetes risk based on clinical parameters, which may help identify individuals who could benefit from further medical evaluation. In addition, the automated notification mechanism promotes consistent glucose monitoring, which is essential for maintaining

glycemic control and preventing long-term complications associated with diabetes. Such systems could complement traditional healthcare services by providing patients with accessible digital tools that support proactive disease management and facilitate communication between patients and healthcare professionals.

Future research could address these limitations by training and validating the model using larger and more diverse clinical datasets. Incorporating additional features related to lifestyle factors, medical history, and wearable sensor data could further improve prediction accuracy and model robustness. Moreover, the integration of additional explainable artificial intelligence techniques (e.g., local interpretable model-agnostic explanations or partial dependence plots) could enhance the transparency and interpretability of the predictive model, allowing both patients and healthcare professionals to better understand the factors influencing prediction outcomes. Finally, extending the system with advanced clinical decision-support functionalities and integrating additional health monitoring modules could further increase its applicability in real-world healthcare environments.

7. Conclusions

This paper presented a mobile-based system for diabetes monitoring and prediction that integrates ML techniques with mobile health technologies. The proposed solution combines continuous health monitoring with predictive analytics, enabling users to record and analyze important health parameters such as glucose levels, insulin injections, and physical activity through a mobile application. By integrating monitoring functionalities with ML-based risk prediction, the system provides users with a comprehensive digital tool that supports both self-management and early awareness of potential diabetes risk.

The experimental evaluation demonstrated that the RF classifier achieved the highest AUC among the evaluated algorithms, indicating superior discrimination capability across different classification thresholds. These results confirm the effectiveness of ensemble learning methods for diabetes prediction tasks when applied to clinical datasets such as the PIMA Indians Diabetes Dataset. Integrating the predictive model into a mobile health platform enables real-time risk assessment while simultaneously allowing users to maintain a structured history of their health data.

An additional contribution of the proposed system is the implementation of an automated notification mechanism that reminds users to record glucose measurements when no data have been logged during the day. This feature encourages consistent self-monitoring behavior, which is widely recognized as an essential component of effective diabetes management. By promoting regular monitoring and providing predictive insights, the proposed system has the potential to support patients in maintaining better glycemic control and improving long-term health outcomes.

Future work will focus on extending the capabilities of the proposed platform by incorporating additional intelligent functionalities. These may include OCR-based medical report analysis, personalized dietary recommendation systems, and the integration of additional physiological data obtained from wearable devices. Furthermore, future research will involve training and validating the predictive models using larger and more diverse clinical datasets in order to improve generalization and predictive performance. Expanding the prediction framework to support the monitoring and early detection of other chronic diseases may also enhance the applicability of the proposed system in broader digital health environments.

Overall, the proposed system demonstrates the potential of integrating ML and mobile health technologies to support proactive diabetes management. By combining predictive analytics with continuous monitoring, such solutions can contribute to earlier disease detection, improved patient awareness, and the development of more personalized digital

healthcare systems. Although additional external clinical validation is required before deployment in routine clinical practice, the proposed framework provides a solid foundation for future intelligent mobile health systems supporting diabetes management.

Author Contributions: Conceptualization, A.F. and A.P.; methodology, A.F., A.P. and A.C.; software, A.F.; validation, A.F., A.C. and D.I.G.; formal analysis, A.F.; investigation, A.F. and A.C.; resources, A.P., D.I.G. and H.V.; data curation, A.F.; writing—original draft preparation, A.F.; writing—review and editing, A.P., A.C., D.I.G. and H.V.; visualization, A.F.; supervision, A.P., D.I.G. and H.V.; project administration, A.P., and A.F. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The dataset used for this study is publicly available on the Kaggle (accessed on 9 July 2026) platform at: <https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database>.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ACC	Accuracy
AUC	Area Under the Receiver Operating Characteristic Curve
DT	Decision Tree
F1	Harmonic Mean of Precision and Recall
FN	False Negative
FP	False Positive
FPR	False Positive Rate
IoMT	Internet of Medical Things
IoT	Internet of Things
JSON	JavaScript Object Notation
ML	Machine Learning
MVC	Model–View–Controller
PPV	Positive Predictive Value (Precision)
REC	Recall
RF	Random Forest
ROC	Receiver Operating Characteristic
SPC	Specificity
SVM	Support Vector Machine
TN	True Negative
TNR	True Negative Rate
TP	True Positive
TPR	True Positive Rate
XAI	Explainable Artificial Intelligence

References

1. American Diabetes Association. Diabetes Technology: Standards of Medical Care in Diabetes—2020. *Diabetes Care* **2020**, *43*, S77–S88. [[CrossRef](#)] [[PubMed](#)]
2. Alserhani, M.F.; Alrasheed, F.; Al Qamshah, A.M.; Alkharashi, Y.; Aljarallah, M.F.; Alarifi, F.F.; Althaqeb, E.; Alrasheed, M.I.; Alkharashi, L.M.; Al Eidan, E. Effectiveness of Self-Management of Blood Glucose in Improving Glycemic Control in Patients with Diabetes: A Systematic Review and Meta-Analysis. *Cureus* **2025**, *17*, e99751. [[CrossRef](#)] [[PubMed](#)]

3. Miller, E.M. Using Continuous Glucose Monitoring in Clinical Practice. *Clin. Diabetes* **2020**, *38*, 429–438. . [CrossRef] [PubMed]
4. Russo, G.T.; Andreozzi, F.; Calabrese, M.; Di Bartolo, P.; Di Cianni, G.; Bruno Giorda, C.; Lapice, E.; Manicardi, E.; Giandalia, A.; Lucisano, G.; et al. Role of telemedicine during COVID-19 pandemic in type 2 diabetes outpatients: The AMD annals initiative. *Diabetes Res. Clin. Pract.* **2022**, *194*, 110158. [CrossRef] [PubMed]
5. Fregoso-Aparicio, L.; Noguez, J.; Montesinos, L.; García-García, J.A. Machine Learning and Deep Learning Predictive Models for Type 2 Diabetes: A Systematic Review. *Diabetol. Metab. Syndr.* **2021**, *13*, 148. [CrossRef] [PubMed]
6. Wee, B.F.; Sivakumar, S.; Lim, K.H.; Wong, W.K.; Juwono, F.H. Diabetes Detection Based on Machine Learning and Deep Learning Approaches. *Multimed. Tools Appl.* **2024**, *83*, 24153–24185. [CrossRef]
7. Farran, B.; Channanath, A.; Behbehani, K.; Thanaraj, T. Predictive Models to Assess Risk of Type 2 Diabetes, Hypertension and Comorbidity: Machine-Learning Algorithms and Validation Using National Health Data from Kuwait—A Cohort Study. *BMJ Open* **2013**, *3*, e002457. [CrossRef] [PubMed]
8. Kodama, S.; Fujihara, K.; Horikawa, C.; Kitazawa, M.; Iwanaga, M.; Kato, K.; Watanabe, K.; Nakagawa, Y.; Matsuzaka, T.; Shimano, H.; et al. Predictive Ability of Current Machine Learning Algorithms for Type 2 Diabetes Mellitus: A Meta-Analysis. *J. Diabetes Investig.* **2022**, *13*, 900–908. [CrossRef] [PubMed]
9. Qin, Y.; Wu, J.; Xiao, W.; Wang, K.; Huang, A.; Liu, B.; Yu, J.; Li, C.; Yu, F.; Ren, Z. Machine Learning Models for Data-Driven Prediction of Diabetes by Lifestyle Type. *Int. J. Environ. Res. Public Health* **2022**, *19*, 15027. [CrossRef] [PubMed]
10. Alshari, H.; Odabas, A. Machine Learning Model to Diagnose Diabetes Type 2 Based on Health Behavior. *Gazi Univ. J. Sci.* **2022**, *35*, 834–852. [CrossRef]
11. Iftikhar, K.; Javaid, N.; Ahmed, I.; Alrajeh, N. A Novel Explainable Deep Learning Framework for Accurate Diabetes Mellitus Prediction. *Appl. Sci.* **2025**, *15*, 9162. [CrossRef]
12. Dashdondov, K.; Lee, S.; Erdenebat, M. Enhancing Diabetes Prediction and Prevention through Mahalanobis Distance and Machine Learning Integration. *Appl. Sci.* **2024**, *14*, 7480. [CrossRef]
13. Wang, X.; Shu, W.; Du, J.; Du, M.; Wang, P.; Xue, M.; Zheng, H.; Jiang, Y.; Yin, S.; Liang, D.; et al. Mobile Health in the Management of Type 1 Diabetes: A Systematic Review and Meta-Analysis. *BMC Endocr. Disord.* **2019**, *19*, 21. [CrossRef] [PubMed]
14. Makroum, M.A.; Adda, M.; Bouzouane, A.; Ibrahim, H. Machine Learning and Smart Devices for Diabetes Management: Systematic Review. *Sensors* **2022**, *22*, 1843. [CrossRef] [PubMed]
15. Joeliantina, A.; Proboningsih, J.; Anugrahini, H.N.; Mejilla, J.L. The Behavior of Patients with Type 2 Diabetes Mellitus in Monitoring Blood Glucose Levels and Foot Care: A Cross-Sectional Community-Based Study. *Int. J. Adv. Health Sci. Technol.* **2022**, *2*, 104–109. [CrossRef]
16. Mensah, G.P.; van Rooyen, D.R.M.; Ten Ham-Baloyi, W. Nursing Management of Gestational Diabetes Mellitus in Ghana: Perspectives of Nurse-Midwives and Women. *Midwifery* **2019**, *71*, 19–26. [CrossRef] [PubMed]
17. Li, J.; Me, R.C.; Ahmad, F.; Zhu, Q. Investigating the Application of IoT Mobile Apps and Healthcare Services for Diabetic Elderly: A Systematic Review. *PLoS ONE* **2025**, *20*, e0321090. [CrossRef] [PubMed]
18. Mastoli, M.; Pol, U.; Patil, D. Machine Learning Model for Prediction of Diabetes Mellitus. *Int. J. Recent Technol. Eng.* **2020**, *8*, 2376–2381. [CrossRef]
19. Kurasawa, H.; Hayashi, K.; Fujino, A.; Takasugi, K.; Haga, T.; Waki, K.; Noguchi, T.; Ohe, K. Machine-Learning-Based Prediction of a Missed Scheduled Clinical Appointment by Patients with Diabetes. *J. Diabetes Sci. Technol.* **2015**, *10*, 730–736. [CrossRef] [PubMed]
20. Ahmed, N.; Ahammed, R.; Islam, M.M.; Uddin, M.A.; Akhter, A.; Talukder, M.A.; Paul, B.K. Machine Learning Based Diabetes Prediction and Development of Smart Web Application. *Int. J. Cogn. Comput. Eng.* **2021**, *2*, 229–241. [CrossRef]
21. Chou, C.-Y.; Hsu, D.-Y.; Chou, C.-H. Predicting the Onset of Diabetes with Machine Learning Methods. *J. Pers. Med.* **2023**, *13*, 406. [CrossRef] [PubMed]
22. Pima Indians Diabetes Dataset. Available online: <https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database> (accessed on 16 June 2026).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.