**HACK4BENGAL**

**Team: Byte Breakers**

**Project : Arogya Marg**

**Problem Statement :** Bridging the gap in emergency medical services and regular health checkups in rural India through telemedicine kiosks equipped with machine learning diagnostics.

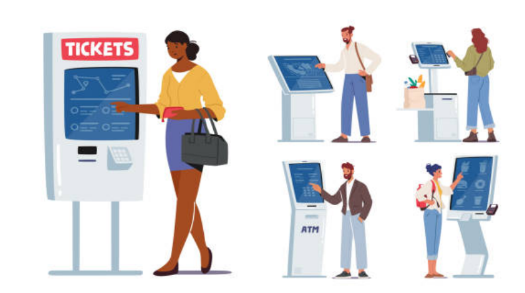
**Problem Statement in Brief:**

In rural India, inadequate transport systems and medical services, combined with a lack of health education and awareness, result in significant challenges for timely access to emergency medical care and regular health checkups.

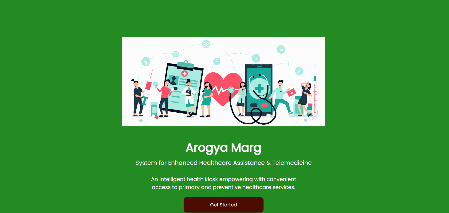
 

**Solution in Brief:**

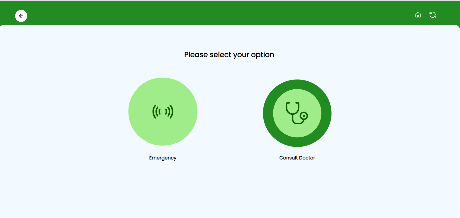
Our telemedicine kiosks enable users to either book appointments for checkups or receive emergency remedies. Equipped with machine learning models, the kiosks can detect common diseases such as diabetes, skin diseases, heart diseases, cataracts, and general health issues, providing instant predictions. For selected cases, users are redirected to e-Sanjeevani for appointments, ensuring timely access to doctors without the need to wait in long queues

.

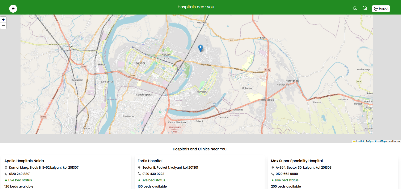
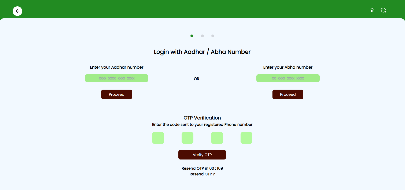
**Frontend Workflow:**



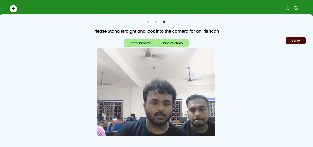
Tap to get started



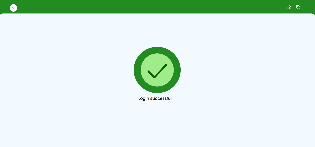
Select Emergency or Consult Doctor

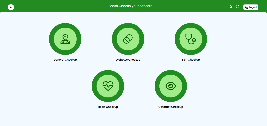
Live Bed Availability & Nearby Hospitals Authentication



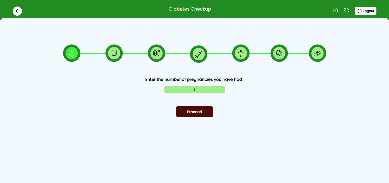
Capture Image



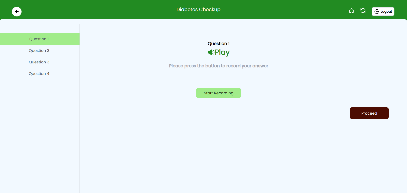
Log In Successful



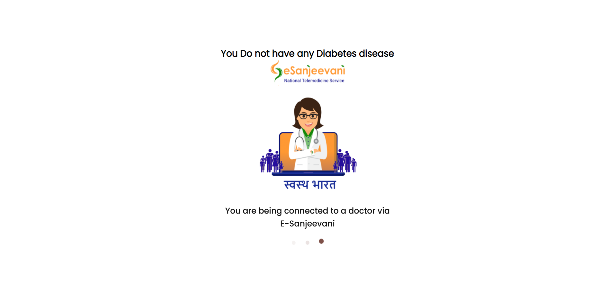
Choose your Concern



Diabetes Checkup



FAQ from users



Directed to e-Sanjeevani

**Frontend Technology :**

In Frontend technologies we have used **React.js** with the help of **Tailwind CSS** .

### Backend for Telemedicine Kiosk Project

**Accuracy and Precision:**

* **Accuracy** measures the proportion of true results (both true positives and true negatives) among the total number of cases examined. It's a key metric in our project to ensure that the machine learning model correctly identifies both diseased and non-diseased individuals.
* **Precision** measures the proportion of true positives among all positive results. High precision is crucial in our project to minimize false positives, ensuring that users are not incorrectly diagnosed and unnecessarily alarmed.

**Importance in Our Project:** Accuracy ensures that our telemedicine kiosk provides reliable disease predictions, which is vital for gaining user trust and ensuring effective medical intervention. Precision is equally important to avoid unnecessary stress and incorrect treatment plans for users.

**Machine Learning Algorithms Used**

**Random Forest Classifier:** The Random Forest classifier is a powerful machine learning algorithm known for its effectiveness in various predictive tasks, including diabetes prediction

**Specific to Diabetes Prediction:**

**Accuracy:** Random Forests tend to achieve high accuracy in diabetes prediction tasks due to their ability to capture complex relationships between input features (e.g., glucose levels, BMI, age) and the target variable (diabetes diagnosis). Here The model achieved an accuracy of 88% on the test dataset this means that 88% of the total predictions (both diabetic and non-diabetic) made by the model are correct.

**Precision:** They often exhibit good precision by minimizing false positives and false negatives, crucial in medical diagnostics where accurate prediction of diabetes cases is essential.The model achieved a precision of 92% on the test dataset. This means that out of all the predictions made by the model where it identified individuals as diabetic, 92% were actually diabetic. High precision indicates a low rate of false positives.

The Random Forest classifier is favored for diabetes prediction and other tasks because of its ensemble nature, reduced overfitting, feature importance assessment, and robustness. These characteristics contribute to its superior performance in terms of accuracy and precision compared to many other machine learning algorithms, particularly in complex datasets with multiple predictors.

**Model Deplementation using Streamlit:** In the context of deploying a diabetes prediction model (or any machine learning model), Streamlit would typically be used for creating a user-friendly front-end interface to interact with the model predictions. The actual deployment would involve a backend framework handling requests and serving predictions from the model. This separation of concerns allows for a more scalable and maintainable architecture for deploying machine learning applications.

**Benefits of Using Streamlit**:

**Rapid Prototyping:** Streamlit allows for quick prototyping and visualization of machine learning models, making it easier to iterate and test different features and parameters. **Interactive UI:** It provides a user-friendly interface for exploring model predictions and results, which can be crucial for stakeholders and end-users in understanding and validating the model's outputs.

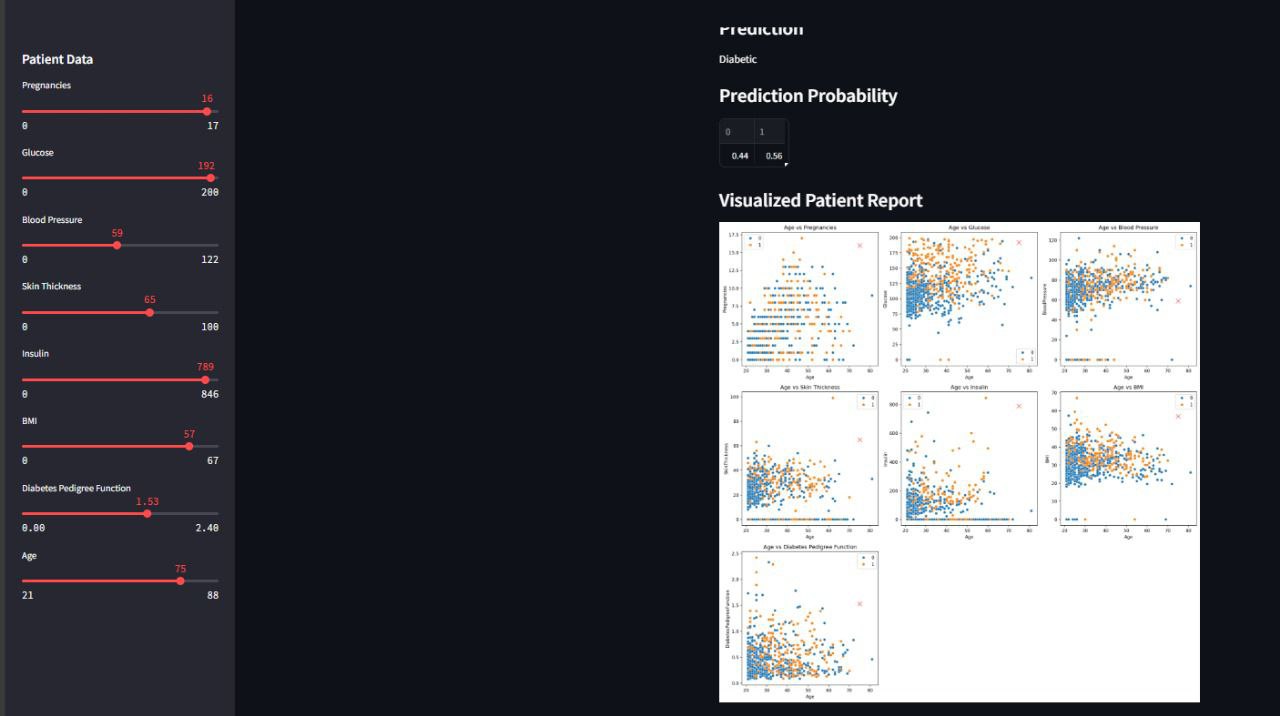
**Workflow for Predicting Diabetes Using Machine Learning Models:**

1. **Define the Problem**
   * **Objective:** Predict diabetes (binary classification).
2. **Data Collection**
   * **Source:** Use datasets like the Pima Indians Diabetes Database.
3. **Data Preprocessing**
   * **Clean Data:** Handle missing values and outliers.
   * **Feature Selection:** Choose relevant features.
   * **Normalization:** Scale features.
   * **Train-Test Split:** Split data (e.g., 80% training, 20% testing).
4. **Exploratory Data Analysis (EDA)**
   * **Visualize:** Plot data distributions and relationships.
   * **Analyze:** Check feature correlations.
5. **Model Selection**
   * **Algorithms:** Random Forest Classifiers.
6. **Model Training**
   * **Fit Models:** Train each model on the training set.
   * **Hyperparameter Tuning:** Optimize using Grid Search or Random Search.
7. **Model Evaluation**
   * **Metrics:** Use accuracy, precision, recall, F1-score, AUC-ROC.
   * **Cross-Validation:** Perform cross-validation to ensure the model generalizes well to unseen data
8. **Model Comparison**
   * **Compare:** Evaluate models based on performance metrics.
   * **Select:** Choose the best model.
9. **Model Deployment**
   * **Save Model:** Use joblib or pickle.
   * **Deploy:** Integrate into a web application or API.
10. **Monitoring and Maintenance**

* **Monitor:** Track performance on new data.
* **Retrain:** Update model periodically with new data

**Generative Output Report:**

We can instantly predict the status of disease regarding patient’s condition.



* **Output Statement**: **The Patient is Diabetic.**

**Conclusion and Future Scope**

Our telemedicine kiosk project aims to bridge the gap in emergency medical services and regular health checkups in rural India. By leveraging machine learning algorithms and integrating with platforms like e-Sanjeevani, we provide a solution that ensures timely and accurate medical care for underserved populations.

**Future Scope:**

1. **Expanded Disease Detection**: Incorporate additional machine learning models to detect a broader range of diseases, including mental health conditions and infectious diseases.

2. **Teleconsultation Services:** Integrate video conferencing capabilities for remote consultations with specialists, enhancing the telemedicine experience.

3. **Health Education:** Develop educational modules within the kiosk to raise awareness about preventive health measures, vaccination schedules, and healthy lifestyle practices.

4. **Wearable Device Integration:** Incorporate data from wearable health devices to provide continuous monitoring and real-time health updates to users.

5. **Community Health Analytics:** Utilize aggregated data to generate community health reports, helping local health authorities to identify and address prevalent health issues more effectively.

**Acknowledgements:**

We would like to express our sincere gratitude to the organizers of Hack4Bengal for providing us with this incredible opportunity to present our idea. Their support and platform have enabled us to develop a solution that we believe will make a significant impact on the healthcare system in rural India. Thank you for fostering innovation and encouraging us to think creatively to solve real-world problems.