**Diabetes Prediction System**

**Team Members:**

1. **Hala Mohammad Omar Ali**
2. **May Talal Albalakha**
3. **Bushra Monim Saaideh**
4. **Razan Saleh Abugnaim**

**Project Overview:**

**This project aims to build a Diabetes Prediction System that predicts whether a person is at risk of diabetes based on medical indicators using a Machine Learning model. The system includes a web interface developed with Flask, allowing users to input their data and receive predictions instantly.**

**Dataset:**

* **Name: Pima Indians Diabetes Dataset**
* **Source: Public Dataset (available on Kaggle and other repositories)**
* **Features:**
  + **Pregnancies**
  + **Glucose**
  + **Blood Pressure**
  + **Skin Thickness**
  + **Insulin**
  + **BMI (Body Mass Index)**
  + **Diabetes Pedigree Function**
  + **Age**
  + **Outcome (Target Variable: 0 = Non-Diabetic, 1 = Diabetic)**

**Tools & Technologies Used:**

* **Python**
* **Pandas & NumPy: Data Handling & Cleaning**
* **Scikit-Learn: Model Training & Evaluation**
* **Flask: Web Development**
* **HTML & CSS: Frontend UI**

**Project Workflow:**

**1. Data Processing & Cleaning**

* **Imported the dataset using Pandas.**
* **Replaced invalid zero values in critical columns (Glucose, Blood Pressure, Skin Thickness, Insulin, BMI) with mean values using SimpleImputer.**

**2. Feature Selection**

* **Features: All columns except Outcome.**
* **Label: Outcome column.**

**3. Splitting Data**

* **Dataset split into 80% Training and 20% Testing using train\_test\_split from Scikit-Learn.**

**4. Model Training**

* **Trained multiple models:**
  + **Logistic Regression**
  + **Decision Tree**
  + **Random Forest**
  + **Support Vector Machine (SVM)**
* **The Random Forest Classifier was chosen due to its balanced performance in Accuracy and Recall.**

**5. Model Evaluation**

* **Metrics evaluated:**
  + **Accuracy**
  + **Precision**
  + **Recall**
  + **F1-Score**

**6. Saving the Model**

* **The best-performing model (Random Forest) was saved using Joblib as diabetes\_model.pkl.**

**Web Application Development**

**Backend:**

* **Built with Flask.**
* **Loads the trained model (diabetes\_model.pkl).**
* **Receives user input via web form, processes the data, and predicts diabetes risk.**

**Frontend:**

* **Developed using HTML and CSS.**
* **A simple form where users can input:**
  + **Pregnancies**
  + **Glucose**
  + **Blood Pressure**
  + **Skin Thickness**
  + **Insulin**
  + **BMI**
  + **Diabetes Pedigree Function**
  + **Age**
* **Displays the prediction:**
  + **"High Risk of Diabetes" or "Low Risk of Diabetes"**

**Frontend Enhancements:**

* **The web interface was later enhanced with a modern, responsive design.**
* **Medical context tips were added next to each input field to improve usability and user awareness.**
* **Visual indicators and styled result alerts distinguish between high and low risk.**

**Future Enhancements**

* **Improve frontend design with advanced CSS/Bootstrap.**
* **Add user authentication to store previous predictions.**
* **Use more advanced ML models or Hyperparameter tuning.**
* **Deploy on cloud platforms like Heroku or Render.**

**Conclusion**

**This project showcases the integration of Machine Learning with Web Development to build a predictive system that is both functional and user-friendly, with an enhanced interface to guide the user during data input.**