**AI Semester Project Report**

**Mobile Application**

**Topic: AI Based Diabetes Predictor**

**GROUP MEMBERS:**

**1.Ammad Khan. (2212105).**

**2.Muhammad Taha. (2212123).**

**3.Muhammad Naqi Askari. (2212132).**

****

**Diabetes Prediction System**

**1. Executive Summary**

The Diabetes Prediction System is a comprehensive full-stack web application aimed at providing early prediction of diabetes risk using machine learning. It enables users to input medical parameters and receives real-time risk assessment, helping in early diagnosis and preventive care. The system is built with a Flask-based backend and a React Native frontend using Expo, and it integrates securely with a PostgreSQL database.

**2. Objectives**

* Develop a secure, user-friendly diabetes prediction platform.
* Apply logistic regression for accurate prediction based on medical data.
* Ensure user data security through encryption and secure API practices.
* Create a scalable frontend for mobile accessibility using React Native.

**3. Technical Architecture**

**3.1 Backend**

* Language: Python
* Framework: Flask with Flask-CORS
* Authentication: Bcrypt password hashing
* Database: PostgreSQL (hosted via Neon.tech)
* Model: Logistic Regression (trained using scikit-learn)
* APIs: RESTful API endpoints with structured response handling

**3.2 Frontend**

* Framework: React Native with Expo
* Language: TypeScript
* Design Pattern: Component-based with hooks and constants
* Mobile Compatibility: Fully responsive mobile design
* Asset Handling: Centralized static asset management

**4. Core Functionalities**

**4.1 Machine Learning Model**

Algorithm: Logistic Regression

Input Features:

* Pregnancies
* Glucose
* Blood Pressure
* Skin Thickness
* Insulin
* BMI
* Diabetes Pedigree Function
* Age

Model Saving: Serialized using pickle as model.pkl

**4.2 User Authentication**

Registration:

* Email, password, and personal details storage
* Duplicate email prevention

Login:

* Bcrypt-hashed password verification
* Session handling and user authentication

**4.3 API Endpoints**

| Endpoint | Method | Description |
| --- | --- | --- |
| /Predict | POST | Predicts diabetes risk |
| /Signup | POST | Registers a new user |
| /Login | POST | Authenticates a user |
| / | GET | Health check endpoint |

**5. Security Considerations**

* Bcrypt: Password encryption and salting
* CORS: Enabled for secure cross-origin communication
* SQL Injection Prevention: Parameterized queries
* Validation: Input sanitization
* Database: SSL-enabled PostgreSQL access
* Logging: Centralized error handling and logging

**6. Project Structure**

├── backend/

│ ├── app.py # Main Flask app

│ ├── train\_model.py # ML model training

│ ├── data.csv # Dataset for training

│ ├── model.pkl # Saved model

│ └── requirements.txt # Python dependencies

└── frontend/

├── app/ # Navigation and routing

├── components/ # Reusable components

├── hooks/ # Custom hooks

├── constants/ # Constant values (colors, config)

├── assets/ # Static files (images, fonts)

└── \*.tsx # Screens and utilities

**7. Development Tools & Dependencies**

**7.1 Backend**

* Python 3.x
* Flask
* Flask-CORS
* pandas
* numpy
* scikit-learn
* psycopg2
* bcrypt

**7.2 Frontend**

* React Native
* Expo
* TypeScript
* ESLint, Prettier
* React Navigation
* Axios

**8. Limitations**

* The model uses logistic regression only, limiting complex pattern detection.
* No integration of historical user data for analysis.
* Frontend lacks unit testing and input validations.

**9. Conclusion**

The Diabetes Prediction System is a robust, full-stack healthcare application that provides a practical tool for assessing diabetes risk using machine learning. It integrates secure backend operations with an accessible mobile frontend. With future scalability, it can be enhanced into a powerful medical tool for both users and healthcare professionals.

**10. Appendix**

**A. Dataset Overview (data.csv)**

| Feature | Description |
| --- | --- |
| Pregnancies | Number of pregnancies |
| Glucose | Plasma glucose concentration |
| BloodPressure | Diastolic blood pressure (mm Hg) |
| SkinThickness | Triceps skinfold thickness (mm) |
| Insulin | 2-Hour serum insulin (mu U/ml) |
| BMI | Body Mass Index |
| DiabetesPedigreeFunction | Diabetes pedigree function |
| Age | Age (years) |
| Outcome | Class variable (0 or 1) |

**B. Sample API Request (POST /predict)**

{

"pregnancies": 2,

"glucose": 120,

"blood\_pressure": 70,

"skin\_thickness": 20,

"insulin": 79,

"bmi": 25.6,

"diabetes\_pedigree": 0.351,

"age": 29

}

**C. Sample API Response**

{

"prediction": "Positive"

}

**D. Results**

 

 