

## MỤC LỤC

<b>CHƯƠNG 1 : TỔNG QUAN</b>	<b>1</b>
1.1 Lý do chọn đề tài	1
1.2 Mục tiêu	1
1.3 Đối tượng	2
1.4 Phạm vi nghiên cứu	3
1.5 Đặt vấn đề	4
1.6 Mục đích	5
<b>CHƯƠNG 2 : NGHIÊN CỨU LÝ THUYẾT</b>	<b>6</b>
2.1 Tổng quan về bệnh tiểu đường	6
2.1.1 Khái niệm	6
2.1.2 Phân loại bệnh tiểu đường	7
2.1.3 Nguyên nhân và yếu tố nguy cơ của bệnh tiểu đường	9
2.1.4 Các chỉ số y tế thường liên quan	12
2.1.5 Biểu chứng của bệnh tiểu đường	14
2.1.6 Tác động xã hội và chi phí y tế của bệnh tiểu đường	16
2.1.7 Vai trò của công nghệ trong phòng ngừa bệnh tiểu đường	18
2.2 Một số công trình và sản phẩm tương tự	19
2.2.1 Các nghiên cứu và mô hình học thuật nổi bật	20
2.2.2 Khảo sát các sản phẩm y tế ứng dụng thực tế	21
2.2.3 Hạn chế của các nghiên cứu và sản phẩm hiện tại	23
2.2.4 Định hướng lý thuyết từ khảo sát	24
2.2.5 Tính mới trong cách tiếp cận lý thuyết	26
2.2.6 Khả năng ứng dụng thực tiễn	27
2.3 Cơ sở lý thuyết về học máy trong dự đoán bệnh tiểu đường	28
2.3.1 Khái niệm về học máy (Machine Learning)	28
2.3.2 Phân loại các mô hình học máy	28
2.3.3 Bài toán phân loại nhị phân	30
2.3.4 Thuật toán Logistic Regression	30
2.3.5 Các chỉ số đánh giá mô hình	32
2.4 Tiền xử lý dữ liệu trong học máy	32
2.4.1 Làm sạch dữ liệu (Data Cleaning)	32

2.4.2 Xử lý giá trị thiếu (Missing Value Handling).....	32
2.4.3 Chuẩn hóa dữ liệu (Normalization / Standardization) .....	33
2.4.4 Mã hóa nhãn (Label Encoding / One-hot Encoding).....	33
2.4.5 Phân chia tập huấn luyện và kiểm tra .....	33
<b>2.5 Ngôn ngữ lập trình Python.....</b>	<b>34</b>
2.5.1 Giới thiệu .....	34
2.5.2 Các thư viện của Python .....	35
2.5.3 Ưu điểm của Python trong triển khai học máy .....	36
2.5.4 Nhược điểm của Python trong triển khai học máy cho đề tài.....	37
2.5.5 Ứng dụng của Python .....	38
<b>2.6 ReactJS.....</b>	<b>39</b>
2.6.1 Lý do chọn ReactJS .....	39
2.6.2 Các thư viện hỗ trợ sử dụng trong ReactJS .....	40
2.6.3 Ưu điểm của ReactJS .....	41
2.6.4 Nhược điểm của ReactJS .....	41
2.6.5 Ứng dụng của ReactJS .....	41
<b>2.7 Hệ quản trị cơ sở dữ liệu MySQL.....</b>	<b>43</b>
2.7.1 Khái niệm về hệ quản trị cơ sở dữ liệu .....	43
2.7.2 Giới thiệu tổng quan về MySQL .....	44
2.7.3 Các đặc điểm nổi bật của MySQL .....	44
2.7.4 Kiến trúc và mô hình dữ liệu trong MySQL.....	45
2.7.5 Ưu điểm và nhược điểm của MySQL.....	45
<b>CHƯƠNG 3 : HIỆN THỰC HÓA NGHIÊN CỨU .....</b>	<b>47</b>
<b>3.1 Tổng quan về hệ thống.....</b>	<b>47</b>
3.1.1 Mô tả hệ thống .....	47
3.1.2 Kiến trúc hệ thống.....	48
3.1.3 Thành phần hệ thống.....	48
3.1.4 Quy trình hoạt động .....	49
<b>3.2 Mô tả chức năng hệ thống .....</b>	<b>49</b>
3.2.1 Giao diện người dùng .....	49
3.2.2 Quản lý tài khoản .....	50
3.2.3 Nhập thông tin y tế.....	50

3.2.4 Dự đoán và hiển thị kết quả .....	50
3.2.5 Lưu và xem lại lịch sử dự đoán.....	50
<b>3.3 Quy trình xử lý dữ liệu y tế .....</b>	<b>51</b>
3.3.1 Giới thiệu tập dữ liệu sử dụng .....	51
3.3.2 Các thuộc tính chính trong tập dữ liệu.....	51
3.3.1 Các bước xử lý dữ liệu.....	52
<b>3.4 Xây dựng và huấn luyện học máy.....</b>	<b>53</b>
3.4.1 Lý do chọn Logistic Regression .....	53
3.4.2 Mô hình Logistic Regression trong scikit-learn .....	53
3.4.3 Lưu mô hình bằng joblib.....	53
3.4.4 Đánh giá mô hình.....	54
3.4.5 Nhận xét về hiệu quả mô hình .....	54
<b>3.5 Thiết kế cơ sở dữ liệu .....</b>	<b>54</b>
3.5.1 Lý do chọn MySQL .....	54
3.5.2 Mô tả các bảng dữ liệu.....	54
3.5.3 Mô tả thuộc tính của các bảng .....	55
<b>3.6 Cài đặt và chạy môi trường Python.....</b>	<b>56</b>
3.6.1 Cài đặt Python.....	56
3.6.2 Cài đặt các thư viện cần thiết.....	57
3.6.3 Chạy Flask API.....	57
<b>3.7 Cài đặt và triển khai giao diện ReactJS .....</b>	<b>57</b>
3.7.1 Tạo project ReactJS .....	57
3.7.2 Cài đặt các thư viện cần thiết.....	58
3.7.3 Chạy thử giao diện.....	58
<b>3.8 Cài đặt XAMPP cho MySQL .....</b>	<b>59</b>
3.8.1 Cài đặt XAMPP .....	59
3.8.2 Truy cập phpMyAdmin .....	59
3.8.3 Tạo cơ sở dữ liệu.....	60
3.8.4 Kiểm tra kết nối với Python.....	60
<b>CHƯƠNG 4 : KẾT QUẢ NGHIÊN CỨU .....</b>	<b>61</b>
<b>4.1 Giao diện trang chủ.....</b>	<b>61</b>
<b>4.2 Giao diện đăng nhập và đăng ký .....</b>	<b>62</b>

4.3 Giao diện trang dự đoán .....	66
4.4 Giao diện kết quả dự đoán .....	67
4.5 Giao diện lịch sử dự đoán .....	69
4.6 Chức năng lọc danh sách lịch sử theo thời gian .....	71
<b>CHƯƠNG 5 : KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....</b>	<b>73</b>
5.1 Kết luận .....	73
5.2 Hướng phát triển.....	73
5.2.1 Hướng phát triển về mặt kỹ thuật .....	74
5.2.2 Hướng phát triển về chức năng người dùng .....	75
5.2.3 Hướng phát triển theo hướng cộng đồng – y tế .....	75
<b>PHỤ LỤC .....</b>	<b>77</b>
<b>DANH MỤC TÀI LIỆU THAM KHẢO .....</b>	<b>78</b>

## LỜI MỞ ĐẦU

Trong những năm gần đây, sự phát triển mạnh mẽ của công nghệ thông tin đã mở ra nhiều hướng tiếp cận mới trong lĩnh vực y tế, đặc biệt là trong việc chẩn đoán và dự đoán bệnh tật dựa trên dữ liệu. Với sự hỗ trợ của trí tuệ nhân tạo và học máy, các mô hình phân tích dữ liệu đã dần khẳng định vai trò quan trọng trong việc hỗ trợ các bác sĩ và chuyên gia y tế đưa ra quyết định chính xác và kịp thời. Trong bối cảnh đó, bệnh tiểu đường là một trong những bệnh mãn tính nguy hiểm và có tỉ lệ mắc ngày càng cao, cũng được quan tâm đặc biệt trong nghiên cứu ứng dụng công nghệ để phát hiện sớm và phòng ngừa hiệu quả.

Đề tài “Xây dựng ứng dụng phân tích dữ liệu y tế để dự đoán bệnh tiểu đường” được thực hiện với mong muốn góp phần nhỏ vào việc ứng dụng công nghệ vào chăm sóc sức khỏe cộng đồng. Ứng dụng được xây dựng không chỉ dừng lại ở việc thu thập và xử lý thông tin người dùng, mà còn triển khai mô hình học máy dự đoán nguy cơ mắc bệnh dựa trên các chỉ số y tế phổ biến như tuổi, chỉ số đường huyết, huyết áp, BMI, tiền sử gia đình. Bên cạnh đó, hệ thống còn cung cấp biểu đồ so sánh trực quan, giúp người dùng hiểu rõ hơn về mức độ rủi ro hiện tại của bản thân.

Em hy vọng báo cáo sẽ phần nào thể hiện được sự nỗ lực học tập, nghiên cứu và thực hành trong suốt thời gian qua. Em cũng rất mong nhận được những góp ý chân thành từ quý Thầy Cô để em có thể tiếp tục hoàn thiện và nâng cao hơn nữa năng lực bản thân trong tương lai.

## LỜI CẢM ƠN

Trước khi đi vào nội dung chính của báo cáo khóa luận tốt nghiệp, em xin được gửi lời cảm ơn chân thành đến những người đã đồng hành, hướng dẫn và hỗ trợ em trong suốt quá trình thực hiện đề tài này.

Trước hết, em xin bày tỏ lòng biết ơn chân thành đến quý Thầy, Cô trong Khoa Công nghệ Thông tin – Trường Đại học Trà Vinh đã tận tình giảng dạy, truyền đạt cho em những kiến thức quý báu trong suốt thời gian học tập tại trường. Những nền tảng kiến thức vững chắc ấy chính là hành trang quan trọng giúp em tự tin thực hiện đề tài này.

Đặc biệt, em xin gửi lời cảm ơn sâu sắc đến Thầy Nguyễn Khắc Quốc là người đã trực tiếp hướng dẫn, theo sát và tận tình góp ý cho em trong suốt quá trình làm khóa luận. Sự tận tâm, kiên nhẫn và những lời động viên kịp thời của Thầy là nguồn động lực lớn giúp em vượt qua nhiều khó khăn để hoàn thành đề tài.

Em cũng xin chân thành cảm ơn các bạn bè, anh/chị trong lớp và trong nhóm nghiên cứu đã chia sẻ, hỗ trợ em cả về chuyên môn lẫn tinh thần trong suốt quá trình làm đề tài. Những lần cùng nhau trao đổi ý tưởng, cùng sửa lỗi, cùng thử nghiệm đã giúp em học được rất nhiều điều quý giá.

Cuối cùng, em xin gửi lời cảm ơn sâu sắc đến gia đình thân yêu – những người luôn ở phía sau, âm thầm ủng hộ, động viên và là chỗ dựa vững chắc cho em trong suốt quãng đường học tập và trưởng thành.

Dù còn nhiều thiếu sót và hạn chế, nhưng em đã luôn cố gắng hết mình để hoàn thành đề tài một cách nghiêm túc và trách nhiệm. Em rất mong nhận được sự thông cảm và góp ý chân thành từ quý Thầy Cô và Hội đồng để em có thể hoàn thiện bản thân tốt hơn trong tương lai.

Em xin chân thành cảm ơn!

Đặng Hào Nguyên

[illegible]

**NHẬN XÉT**  
(Của giảng viên hướng dẫn trong đề án, khoá luận của sinh viên)

.....

.....

.....

.....

.....

.....

.....

**Giảng viên hướng dẫn**  
(ký và ghi rõ họ tên)



## **BẢN NHẬN XÉT ĐỒ ÁN, KHÓA LUẬN TỐT NGHIỆP**

Họ và tên sinh viên: ..... MSSV: .....

Ngành: ..... Khóa: .....

Tên đề tài: .....

.....

.....

Họ và tên Giáo viên hướng dẫn: .....

Chức danh: ..... Học vị: .....

### **NHẬN XÉT**

#### **1. Nội dung đề tài:**

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

#### **2. Ưu điểm:**

.....

.....

.....

.....

#### **3. Khuyết điểm:**

.....

.....

.....

.....

.....

4. Điểm mới đề tài:

.....

.....

.....

.....

.....

5. Giá trị thực trên đề tài:

.....

.....

.....

.....

.....

.....

.....

7. Đề nghị sửa chữa bổ sung:

.....

.....

.....

.....

.....

.....

.....

8. Đánh giá:

.....

.....

.....

.....

Trà Vinh, ngày tháng năm 20...  
Giảng viên hướng dẫn  
(Ký & ghi rõ họ tên)

**NHẬN XÉT**  
**(Của giảng viên chấm trong đồ án, khoá luận của sinh viên)**

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

**Giảng viên chấm**  
(ký và ghi rõ họ tên)

**BẢN NHẬN XÉT ĐỒ ÁN, KHÓA LUẬN TỐT NGHIỆP**  
(*Của cán bộ chấm đồ án, khóa luận*)

Họ và tên người nhận xét: .....

Chức danh: ..... Học vị: .....

Chuyên ngành: .....

Cơ quan công tác: .....

Tên sinh viên: .....

Tên đề tài đồ án, khóa luận tốt nghiệp:

.....  
.....

**I. Ý KIẾN NHẬN XÉT**

1. Nội dung:

.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....

2. Điểm mới các kết quả của đồ án, khóa luận:

.....  
.....  
.....

3. Ứng dụng thực tế:

.....  
.....  
.....  
.....  
.....

(Các câu hỏi của giáo viên phản biện)

### III. KẾT LUẬN

Người nhận xét  
(Ký & ghi rõ họ tên)

## **DANH MỤC CÁC BẢNG, SƠ ĐỒ, HÌNH**

BẢNG 3. 1. MÔ TẢ CÁC THUỘC TÍNH TRONG TẬP DỮ LIỆU .....	51
BẢNG 3. 2. MÔ TẢ CÁC BẢNG DỮ LIỆU .....	54
BẢNG 3. 3. MÔ TẢ THUỘC TÍNH BẢNG USERS.....	55
BẢNG 3. 4. BẢNG MÔ TẢ THUỘC TÍNH BẢNG PREDICTIONS.....	55
HÌNH 2. 1. TỔNG QUAN VỀ BỆNH TIỂU ĐƯỜNG .....	6
HÌNH 2. 2. NGUYÊN NHÂN GÂY BỆNH TIỂU ĐƯỜNG .....	10
HÌNH 2. 3. BIẾN CHỨNG CỦA BỆNH TIỂU ĐƯỜNG .....	15
HÌNH 2. 4. MACHINE LEARNING – HỌC MÁY .....	28
HÌNH 2. 5. MÔ HÌNH HỌC MÁY CÓ GIÁM SÁT .....	29
HÌNH 2. 6. MÔ HÌNH HỌC MÁY KHÔNG GIÁM SÁT .....	29
HÌNH 2. 7. THUẬT TOÁN LOGISTIC REGRESSION .....	31
HÌNH 2. 8. NGÔN NGỮ LẬP TRÌNH PYTHON.....	34
HÌNH 2. 9. CÁC THƯ VIỆN CỦA PYTHON .....	35
HÌNH 2. 10. REACTJS – CÔNG CỤ PHÁT TRIỂN GIAO DIỆN NGƯỜI DÙNG...39	
HÌNH 2. 11. CÁC THƯ VIỆN TRONG REACTJS .....	40
HÌNH 2. 12. MYSQL - HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU QUAN HỆ.....	44
HÌNH 3. 1. GIAO DIỆN NƠI TẢI VỀ ĐỂ CÀI ĐẶT PYTHON.....	56
HÌNH 3. 2. GIAO DIỆN NƠI TẢI VỀ ĐỂ CÀI ĐẶT XAMPP .....	59
HÌNH 4. 1. GIAO DIỆN TRANG CHỦ .....	61
HÌNH 4. 2. GIAO DIỆN HƯỚNG DẪN SỬ DỤNG HỆ THỐNG.....	62
HÌNH 4. 3. GIAO DIỆN ĐĂNG NHẬP .....	63
HÌNH 4. 4. GIAO DIỆN ĐĂNG KÝ .....	63
HÌNH 4. 5. GIAO DIỆN XÁC THỰC MÃ OTP ĐỂ ĐĂNG KÝ .....	64
HÌNH 4. 6. GIAO DIỆN ĐẶT LẠI MẬT KHẨU .....	65
HÌNH 4. 7. GIAO DIỆN XÁC THỰC MÃ OTP.....	65
HÌNH 4. 8. GIAO DIỆN NHẬP MẬT KHẨU MỚI .....	66
HÌNH 4. 9. GIAO DIỆN TRANG CHỨC NĂNG DỰ ĐOÁN .....	67
HÌNH 4. 10. GIAO DIỆN DỰ ĐOÁN TRẢ KẾT QUẢ .....	67

HÌNH 4. 11. GIAO DIỆN XEM KẾT QUẢ DẠNG BIỂU ĐỒ .....	68
HÌNH 4. 12. GIAO DIỆN TRANG LỊCH SỬ DỰ ĐOÁN .....	70
HÌNH 4. 13. GIAO DIỆN XEM LẠI CHI TIẾT LỊCH SỬ .....	70
HÌNH 4. 14. LỌC LỊCH SỬ THEO NGÀY, THÁNG, NĂM.....	71

**KÍ HIỆU CÁC CỤM TỪ VIẾT TẮT**

<b>STT</b>	<b>Chữ viết tắt</b>	<b>Tên tiếng Anh</b>	<b>Tên tiếng Việt</b>
1.	ADA	American Diabetes Association	Hiệp hội Tiểu đường Hoa Kỳ
2.	IDF	International Diabetes Federation	Liên đoàn Đái tháo đường Quốc tế
3.	MODY	Maturity Onset Diabetes of the Young	Đái tháo đường khởi phát sớm ở người trẻ do yếu tố di truyền
4.	WHO	World Health Organization	Tổ chức Y tế Thế giới
5.	AI	Artificial Intelligence	Trí tuệ nhân tạo
6.	ML	Machine Learning	Học máy
7.	NIDDK	National Institute of Diabetes and Digestive and Kidney Diseases	Viện Quốc gia về Tiểu đường, Tiêu hóa và Bệnh thận (Hoa Kỳ)
8.	EHR	Electronic Health Record	Hồ sơ bệnh án điện tử
9.	FDA	Food and Drug Administration	Cục Quản lý Dược & Thực phẩm Hoa Kỳ
10.	RDBMS	Relational Database Management System	Hệ quản trị cơ sở dữ liệu quan hệ
11.	DBMS	Database Management System	Hệ quản trị cơ sở dữ liệu
12.	SMOTE	Synthetic Minority Over-sampling Technique	Kỹ thuật tạo mẫu thiếu số tổng hợp
13.	MRI	Magnetic Resonance Imaging	Chụp cộng hưởng từ



## **CHƯƠNG 1: TỔNG QUAN**

### **1.1 Lý do chọn đề tài**

Trong bối cảnh cuộc sống hiện đại ngày càng bận rộn, bệnh tiểu đường đã trở thành một trong những vấn đề y tế toàn cầu đáng báo động. Theo thống kê của Tổ chức Y tế Thế giới (WHO), số người mắc bệnh tiểu đường đang không ngừng gia tăng, đặc biệt tại các quốc gia đang phát triển, nơi điều kiện chăm sóc sức khỏe còn nhiều hạn chế. Việc phát hiện sớm nguy cơ mắc bệnh có ý nghĩa đặc biệt quan trọng trong công tác phòng ngừa và điều trị kịp thời.

Bệnh tiểu đường (Diabetes Mellitus) là một nhóm bệnh rối loạn chuyển hóa, trong đó lượng đường (Glucose) trong máu tăng cao kéo dài. Nguyên nhân chủ yếu đến từ việc tuyến tụy không sản xuất đủ insulin hoặc cơ thể không sử dụng hiệu quả insulin, một hormone giúp vận chuyển đường từ máu vào tế bào.

Tiểu đường nếu không được phát hiện và điều trị kịp thời có thể gây ra nhiều biến chứng nguy hiểm như mù lòa, suy thận, bệnh tim mạch, đột quỵ, hoặc cắt cụt chi. Đặc biệt, nhiều trường hợp mắc bệnh nhưng không biểu hiện triệu chứng rõ ràng trong giai đoạn đầu, dẫn đến chủ quan và chậm trễ trong việc can thiệp y tế.

Bên cạnh đó, với sự phát triển vượt bậc của công nghệ, đặc biệt là trí tuệ nhân tạo và học máy (Machine Learning), việc ứng dụng các mô hình phân tích dữ liệu y tế để hỗ trợ chẩn đoán, dự đoán bệnh tật đã trở nên phổ biến và hiệu quả. Tuy nhiên, ở Việt Nam, những ứng dụng như vậy vẫn còn hạn chế về mặt tiếp cận và chưa thật sự phổ biến đối với người dân.

Chính vì vậy, em lựa chọn đề tài “Xây dựng ứng dụng phân tích dữ liệu y tế để dự đoán bệnh tiểu đường” với mong muốn vừa vận dụng kiến thức công nghệ đã học, vừa góp phần vào việc nâng cao nhận thức cộng đồng trong việc phòng ngừa căn bệnh nguy hiểm này.

### **1.2 Mục tiêu**

Đề tài được thực hiện với các mục tiêu cụ thể sau:

Xây dựng một hệ thống ứng dụng web thân thiện, cho phép người dùng nhập các chỉ số y tế cá nhân.

Áp dụng mô hình học máy để dự đoán nguy cơ mắc bệnh tiểu đường dựa trên các chỉ số đã nhập.

Trực quan hóa kết quả dự đoán bằng biểu đồ giúp người dùng dễ hiểu và đưa ra quyết định phù hợp.

Tích hợp chức năng lưu lịch sử dự đoán để người dùng có thể theo dõi và so sánh theo thời gian.

Tạo nền tảng ứng dụng có thể mở rộng cho các loại bệnh lý khác trong tương lai.

Để đạt được các mục tiêu trên, đề tài khóa luận sử dụng các công nghệ sau:

Ngôn ngữ lập trình: Python (Flask) cho backend, JavaScript (ReactJS) cho frontend.

Mô hình học máy: Logistic Regression ( sử dụng để huấn luyện mô hình bằng thư viện scikit-learn) kết hợp với xử lý dữ liệu qua pandas và chuẩn hóa bằng StandardScaler.

Cơ sở dữ liệu: MySQL để lưu thông tin người dùng và lịch sử dự đoán.

Giao tiếp frontend-backend: thông qua API sử dụng JSON và giao thức HTTP.

Gửi email xác thực / khôi phục mật khẩu: sử dụng thư viện smtplib và Gmail SMTP Server.

Trực quan hóa dữ liệu: dùng thư viện Recharts trên React để hiển thị biểu đồ cột so sánh các chỉ số.

### **1.3 Đối tượng**

Đề tài tập trung nghiên cứu các thành phần chính liên quan đến quá trình dự đoán bệnh tiểu đường thông qua mô hình học máy, gồm:

Các chỉ số y tế có liên quan đến nguy cơ mắc bệnh tiểu đường như:

- Tuổi, giới tính
- Chỉ số đường huyết (glucose)
- Huyết áp tâm thu, huyết áp tâm trương
- Chỉ số BMI, chiều cao, cân nặng

- Nhịp tim
- Tiền sử bệnh lý trong gia đình như tiểu đường, huyết áp cao, đột quỵ, bệnh tim mạch.

Mô hình học máy Logistic Regression, sử dụng để phân loại nguy cơ mắc bệnh trên nền tảng dữ liệu đầu vào đã được chuẩn hóa.

Quá trình xử lý dữ liệu y tế bao gồm làm sạch, chuẩn hóa, sắp xếp thứ tự đặc trưng, nhằm đảm bảo độ chính xác và khả năng tổng quát của mô hình.

Ứng dụng web với giao diện thân thiện, được phát triển để người dùng nhập thông tin và nhận kết quả dự đoán trực tiếp trên nền tảng trực tuyến.

Hệ thống được xây dựng hướng đến hai nhóm đối tượng chính:

Người dùng phổ thông (không chuyên y tế): có thể tự nhập chỉ số cá nhân để xem nguy cơ mắc bệnh tiểu đường dưới dạng kết luận và biểu đồ minh họa. Hệ thống giúp họ tự theo dõi và nâng cao nhận thức sức khỏe bản thân.

Sinh viên và người nghiên cứu lĩnh vực công nghệ – y tế: có thể sử dụng mô hình như một ví dụ thực tế về ứng dụng trí tuệ nhân tạo trong chăm sóc sức khỏe. Đây là tài nguyên tham khảo hữu ích cho các đề tài nghiên cứu, khóa luận hoặc triển khai sản phẩm thật trong tương lai.

#### **1.4 Phạm vi nghiên cứu**

Phạm vi nghiên cứu của đề tài bao gồm:

Nghiên cứu và ứng dụng mô hình học máy để huấn luyện mô hình từ bộ dữ liệu bệnh tiểu đường.

Hệ thống tập trung vào việc dự đoán nguy cơ mắc bệnh tiểu đường.

Xây dựng hệ thống dựa trên ngôn ngữ lập trình Python với Flask cho backend và ReactJS cho frontend.

Phạm vi đề tài không đi sâu vào khám, chẩn đoán y khoa chính thức mà chỉ mang tính chất hỗ trợ tham khảo và nâng cao nhận thức.

## 1.5 Đặt vấn đề

Trong những năm gần đây, tỷ lệ mắc bệnh tiểu đường đang gia tăng nhanh chóng không chỉ ở người cao tuổi mà còn ở người trẻ tuổi, đặc biệt tại các thành phố lớn – nơi áp lực công việc, chế độ ăn uống và thói quen sinh hoạt thiếu lành mạnh ngày càng phổ biến. Tuy nhiên, do đặc thù là bệnh tiến triển âm thầm, phần lớn bệnh nhân chỉ phát hiện ra khi bệnh đã ở giai đoạn muộn, gây ra nhiều biến chứng nguy hiểm và làm tăng chi phí điều trị.

Một trong những nguyên nhân chính dẫn đến tình trạng này là sự thiếu hụt các công cụ đánh giá nguy cơ bệnh sớm, đặc biệt đối với người không có điều kiện khám sức khỏe định kỳ hoặc không có kiến thức y tế chuyên môn. Trong khi đó, nhiều chỉ số sức khỏe có thể được đo đơn giản tại nhà (tuổi, cân nặng, chiều cao, huyết áp, chỉ số đường huyết), nếu được phân tích đúng cách, hoàn toàn có thể giúp dự đoán sớm nguy cơ mắc bệnh tiểu đường.

Song song đó, trí tuệ nhân tạo (AI) và kỹ thuật học máy (Machine Learning) ngày càng chứng minh hiệu quả vượt trội trong phân tích dữ liệu lớn, phát hiện mô hình, và đưa ra kết quả dự đoán chính xác. Nhiều quốc gia phát triển đã ứng dụng AI vào chẩn đoán bệnh lý từ rất sớm, nhưng ở Việt Nam, các hệ thống như vậy còn hiếm, hoặc khó tiếp cận với người dân bình thường.

Ngoài ra, hiện chưa có nhiều ứng dụng giao diện thân thiện, dễ dùng, và miễn phí giúp người dân tự kiểm tra nguy cơ mắc bệnh ngay tại nhà, đồng thời có thể lưu trữ lịch sử dự đoán để theo dõi theo thời gian. Việc này tạo ra một khoảng trống giữa công nghệ y tế hiện đại và nhu cầu theo dõi sức khỏe cá nhân hàng ngày, đặc biệt trong cộng đồng trẻ tuổi, sinh viên, người làm văn phòng là những đối tượng có nguy cơ tiềm ẩn nhưng thường chủ quan.

Xuất phát từ thực trạng đó, đề tài “Xây dựng ứng dụng phân tích dữ liệu y tế để dự đoán bệnh tiểu đường” với mục tiêu tạo ra một hệ thống kết hợp giữa công nghệ học máy và giao diện web thân thiện, giúp người dùng dễ dàng sử dụng mà không cần hiểu biết sâu về y học hay lập trình. Qua đó, đề tài mong muốn góp phần nhỏ vào việc nâng cao nhận thức, cảnh báo sớm và hỗ trợ cộng đồng tiếp cận với công cụ chăm sóc sức khỏe hiện đại, dễ tiếp cận và hiệu quả.

## 1.6 Mục đích

Đề tài “Xây dựng ứng dụng phân tích dữ liệu y tế để dự đoán bệnh tiểu đường” được thực hiện với mục đích chính là tạo ra một hệ thống ứng dụng web thông minh, thân thiện, giúp người dùng phổ thông có thể tự đánh giá nguy cơ mắc bệnh tiểu đường dựa trên các chỉ số y tế cơ bản.

Đề tài khóa luận thực hiện nhằm các mục đích bao gồm:

Ứng dụng kiến thức đã học vào thực tiễn, đặc biệt là trong lĩnh vực học máy xử lý dữ liệu và lập trình website.

Tạo ra một công cụ hỗ trợ kiểm tra sức khỏe dễ sử dụng, thân thiện với mọi đối tượng, kể cả người không có chuyên môn y tế hay kỹ thuật.

Cảnh báo sớm nguy cơ mắc bệnh tiểu đường, giúp người dùng nâng cao ý thức phòng tránh và chủ động thay đổi lối sống lành mạnh hơn.

Hỗ trợ người nghiên cứu và sinh viên ngành công nghệ tham khảo mô hình ứng dụng thực tế giữa AI và y tế, có thể mở rộng, tùy chỉnh và tích hợp vào các đề tài nghiên cứu khác.

Thúc đẩy tư duy tích hợp đa lĩnh vực: kết hợp giữa công nghệ – y tế – giáo dục – xã hội, nhằm giải quyết vấn đề mang tính cộng đồng một cách sáng tạo và có trách nhiệm.

Tạo tiền đề cho các hệ thống ứng dụng thông minh trong tương lai, không chỉ giới hạn ở bệnh tiểu đường mà còn có thể mở rộng dự đoán các bệnh lý khác như: cao huyết áp, tim mạch, béo phì, đột quỵ.

## CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT

### 2.1 Tổng quan về bệnh tiểu đường

#### 2.1.1 Khái niệm

Theo Hướng dẫn chẩn đoán và điều trị đái tháo đường típ 2 của Bộ Y tế (Quyết định số 3319/QĐ-BYT, 2017), tiểu đường, hay còn gọi là đái tháo đường, là một căn bệnh mãn tính khá phổ biến hiện nay, không chỉ ở người lớn tuổi mà còn bắt đầu xuất hiện ngày càng nhiều ở người trẻ. Về cơ bản, đây là căn bệnh xảy ra khi cơ thể có quá nhiều đường trong máu hay tăng đường huyết kéo dài [2].



Hình 2. 1. Tổng quan về bệnh tiểu đường

Thông thường, sau khi chúng ta ăn uống, phần lớn chất dinh dưỡng sẽ được chuyển hóa thành đường (glucose), rồi đi vào máu. Lúc này, insulin – một loại hormone do tuyến tụy tiết ra sẽ giúp mở đường cho glucose đi vào tế bào để được chuyển hóa thành năng lượng. Tuy nhiên, với người bị tiểu đường, cơ thể không sản xuất đủ insulin, hoặc sản xuất rồi mà lại không sử dụng được hiệu quả. Kết quả là đường bị kẹt lại trong máu, không vào tế bào được dẫn tới đường huyết tăng cao.

Mới nghe thì có vẻ không nghiêm trọng, nhưng nếu tình trạng này kéo dài nhiều tháng hoặc nhiều năm mà không phát hiện, đường trong máu sẽ âm thầm phá hoại các cơ quan như tim, thận, mắt, thần kinh.... Có nhiều trường hợp đến khi phát hiện thì mắt đã mờ, thậm chí phải chạy thận vì biến chứng thận do tiểu đường.

Theo báo cáo của Tổ chức Y tế Thế giới, tính đến năm 2021, trên thế giới có hơn 537 triệu người trưởng thành đang sống chung với bệnh tiểu đường, trong đó rất nhiều người không hề biết mình mắc bệnh. Riêng ở Việt Nam, thống kê của Bộ Y tế cho thấy tỷ lệ mắc tiểu đường đang gia tăng nhanh, đặc biệt ở khu vực thành thị – nơi có lối sống ít vận động, ăn uống nhiều tinh bột và chất béo [6, 43].

Một điều nguy hiểm ở bệnh này là thường không có dấu hiệu rõ ràng ở giai đoạn đầu. Người bệnh vẫn ăn uống bình thường, không sốt, không đau, chỉ đôi khi hơi mệt, khát nước hoặc đi tiểu nhiều.

Chính vì vậy mà tiểu đường được gọi là “căn bệnh thầm lặng”. Người bệnh có thể sống hàng năm mà không biết, cho đến khi bị biến chứng mới phát hiện thì đã muộn.

Tóm lại, tiểu đường là bệnh rối loạn chuyển hóa đường, ảnh hưởng nghiêm trọng đến sức khỏe nếu không được phát hiện sớm và điều trị kịp thời.

### **2.1.2 Phân loại bệnh tiểu đường**

Theo phân loại trong tài liệu của Bộ Y tế (2014), bệnh tiểu đường không chỉ có một dạng duy nhất. Thực tế, nó được chia thành ba loại chính dựa trên cơ chế gây bệnh, thời điểm xuất hiện và đặc điểm người mắc. Việc phân biệt rõ từng loại là rất quan trọng vì mỗi loại sẽ có cách chẩn đoán, điều trị và theo dõi khác nhau. Trong phần này, có lần lượt ba loại tiểu đường phổ biến nhất hiện nay: tuýp 1, tuýp 2 và tiểu đường thai kỳ [1].

#### **Tiểu đường tuýp 1 – Cơ thể không còn sản xuất insulin**

Đây là dạng tiểu đường ít gặp hơn, thường xuất hiện ở trẻ em, thanh thiếu niên hoặc người trẻ tuổi. Trong trường hợp này, cơ thể bị nhầm lẫn, tấn công vào chính tuyến tụy – nơi sản xuất insulin. Khi tế bào beta của tuyến tụy bị phá hủy, người bệnh mất hoàn toàn khả năng sản xuất insulin.

Người bệnh phải tiêm insulin mỗi ngày để sống và duy trì đường huyết ổn định. Nếu không điều trị đúng, đường huyết sẽ tăng rất cao chỉ sau vài ngày. Các triệu chứng thường đến rất nhanh: khát nước nhiều, đi tiểu nhiều, giảm cân đột ngột, mệt mỏi, thậm chí hôn mê.

Theo thống kê của Tổ chức Y tế Thế giới (WHO, 2023), tiểu đường tuýp 1 chiếm khoảng 5–10% tổng số ca bệnh tiểu đường trên toàn cầu [6, 43].

### **Tiểu đường tuýp 2 – Insulin có nhưng không hiệu quả**

Đây là loại phổ biến nhất, chiếm trên 90% tổng số người mắc bệnh tiểu đường. Cơ thể người bệnh vẫn sản xuất insulin, nhưng các tế bào trong cơ thể lại không phản ứng hiệu quả với insulin đó – gọi là tình trạng kháng insulin.

Tiểu đường tuýp 2 thường gặp ở:

- Người trung niên hoặc lớn tuổi (trên 40),
- Người béo phì, ít vận động,
- Người có chế độ ăn nhiều tinh bột, đường và chất béo xấu,
- Hoặc có tiền sử gia đình mắc bệnh.

Điểm đặc biệt của tuýp 2 là bệnh diễn tiến âm thầm. Nhiều người chỉ phát hiện tình cờ khi đi khám sức khỏe định kỳ hoặc có biến chứng nhẹ (mờ mắt, loét chân, mệt mỏi...).

Nếu phát hiện sớm, tuýp 2 có thể kiểm soát tốt bằng thay đổi lối sống, ăn uống điều độ, tập thể dục và dùng thuốc theo chỉ định. Tuy nhiên, nếu để kéo dài, nhiều bệnh nhân tuýp 2 vẫn cần tiêm insulin như tuýp 1.

Theo Hiệp hội Tiểu đường Hoa Kỳ (ADA, 2023), mỗi năm có thêm khoảng 1,5 triệu người được chẩn đoán mắc tiểu đường tuýp 2 chỉ riêng tại Hoa Kỳ [7].

### **Tiểu đường thai kỳ – Xuất hiện trong thời kỳ mang thai**

Tiểu đường thai kỳ là tình trạng rối loạn dung nạp glucose trong giai đoạn mang thai, đặc biệt trong tam cá nguyệt thứ hai hoặc thứ ba. Thường thì trước đó người mẹ không hề có dấu hiệu của tiểu đường.

Nguyên nhân là do:

- Thay đổi nội tiết tố trong thai kỳ làm giảm độ nhạy insulin.
- Nhu cầu chuyển hóa tăng nhanh khi thai lớn, cơ thể mẹ không thích ứng kịp.



Nếu không kiểm soát tốt, tiểu đường thai kỳ có thể gây biến chứng cho cả mẹ và bé:

- Thai nhi quá lớn, dễ sinh mổ.
- Bé sinh ra có nguy cơ hạ đường huyết, béo phì sớm, thậm chí tiểu đường sau này.
- Người mẹ có nguy cơ mắc tiểu đường tuýp 2 sau sinh.

Theo IDF Diabetes Atlas (2021), ước tính có hơn 20 triệu phụ nữ trên toàn thế giới mắc tiểu đường thai kỳ mỗi năm [22].

### **Một số dạng hiếm gặp khác**

Ngoài ba loại chính, y học còn ghi nhận một số thể tiểu đường đặc biệt, ít gặp như:

- MODY (Maturity Onset Diabetes of the Young): do đột biến gen đơn lẻ.
- Tiểu đường do thuốc (Steroid-Induced): thường gặp ở bệnh nhân dùng corticoid lâu ngày.
- Tiểu đường thứ phát: do bệnh lý tuyến tụy hoặc nội tiết.

Những dạng này chiếm tỷ lệ rất nhỏ nhưng cần được phân biệt đúng trong chẩn đoán lâm sàng.

### **2.1.3 Nguyên nhân và yếu tố nguy cơ của bệnh tiểu đường**

Bệnh tiểu đường, đặc biệt là tuýp 2, thường không đến một cách đột ngột. Thay vào đó, là kết quả của quá trình tích lũy nhiều yếu tố nguy cơ trong thời gian dài từ thói quen sống, yếu tố di truyền đến môi trường sống và công việc.

Việc hiểu rõ nguyên nhân và những yếu tố có thể dẫn đến tiểu đường không chỉ giúp phòng tránh mà còn là cơ sở quan trọng để xây dựng mô hình dự đoán bệnh sớm – điều mà đề tài này đang hướng đến.



Hình 2. 2. Nguyên nhân gây bệnh tiểu đường

Dưới đây là các nhóm yếu tố nguy cơ chính được y văn và các tổ chức y tế quốc tế công nhận.

#### **Di truyền và tiền sử gia đình:**

Nếu trong gia đình có người từng mắc bệnh tiểu đường, đặc biệt là bố mẹ ruột hoặc anh chị em ruột, thì nguy cơ của cá nhân cũng sẽ cao hơn rất nhiều. Đây là yếu tố không thể thay đổi nhưng rất quan trọng trong dự đoán

Theo Harvard Medical School (2021), nếu cả cha và mẹ đều mắc bệnh tiểu đường tuýp 2, con cái có nguy cơ cao hơn bình thường gấp 5 đến 6 lần [20].

Người có người thân mắc bệnh còn dễ có những thói quen sống tương tự nhau (ăn uống, vận động), nên càng làm tăng nguy cơ.

#### **Tuổi tác và giới tính:**

- Càng lớn tuổi, nguy cơ mắc bệnh tiểu đường càng tăng. Đặc biệt, từ sau 40 tuổi, cơ thể bắt đầu lão hóa, khả năng kiểm soát glucose kém dần [2].
- Nam giới có xu hướng mắc sớm hơn, trong khi nữ giới sau mãn kinh lại có nguy cơ tăng nhanh do thay đổi hormone và chuyển hóa.

Theo WHO (2023), khoảng 1/3 số ca bệnh tiểu đường được phát hiện sau tuổi 45 [42].

### **Béo phì và thừa cân:**

- Đây là yếu tố nguy cơ lớn nhất và dễ thấy nhất ở phần lớn người bệnh.
- Người có chỉ số BMI từ 25 trở lên được xem là thừa cân; từ 30 trở lên là béo phì.
- Mỡ nội tạng (mỡ quanh gan, ruột) làm tăng tình trạng kháng insulin, khiến glucose khó đi vào tế bào.

Theo thống kê từ International Diabetes Federation (2021), khoảng 80–90% người mắc tiểu đường tuýp 2 là người thừa cân hoặc béo phì [22].

### **Lối sống ít vận động:**

- Lười vận động thể chất làm giảm độ nhạy của tế bào với insulin.
- Dù bạn không thừa cân, nhưng nếu ngồi nhiều (>6–8 tiếng/ngày) mà không tập thể dục thì vẫn dễ bị tiểu đường.

Nhiều người làm văn phòng, công nghệ, học online... đang rơi vào nhóm nguy cơ này [2].

### **Chế độ ăn uống không lành mạnh:**

- Ăn nhiều tinh bột tinh chế (cơm trắng, bánh mì trắng), thực phẩm siêu chế biến, nhiều dầu mỡ, đường → tăng nguy cơ cao.
- Ăn ít rau xanh, chất xơ và trái cây → giảm khả năng kiểm soát đường huyết.
- Thói quen uống nước ngọt, trà sữa, thức uống năng lượng mỗi ngày cũng là nguyên nhân đáng báo động ở giới trẻ.

Theo báo cáo của Bộ Y tế Việt Nam (2021), học sinh – sinh viên ở thành phố có thói quen tiêu thụ đường gần gấp đôi so với khuyến nghị của WHO [3].

### **Stress kéo dài và mất ngủ:**

- Căng thẳng mãn tính làm tăng hormone cortisol, khiến gan giải phóng glucose vào máu nhiều hơn.
- Thiếu ngủ (<6 tiếng mỗi đêm) hoặc rối loạn giấc ngủ làm giảm độ nhạy insulin và gây rối loạn nội tiết.

Một nghiên cứu của Mayo Clinic (2020) cho thấy người mất ngủ thường xuyên có nguy cơ mắc tiểu đường cao hơn người ngủ đủ từ 25–40% [25].

### **Tiền sử các bệnh lý khác:**

Người từng bị tăng huyết áp, rối loạn mỡ máu, hội chứng buồng trứng đa nang (PCOS) hoặc bệnh gan nhiễm mỡ không do rượu (NAFLD) cũng có nguy cơ cao hơn.

Đặc biệt, những người từng được chẩn đoán tiền tiểu đường (Pre-diabetes) nếu không thay đổi lối sống sẽ có 30–50% khả năng chuyển sang tiểu đường tuýp 2 trong vòng 5 năm.

### **2.1.4 Các chỉ số y tế thường liên quan**

Khi nói đến bệnh tiểu đường, đặc biệt là tiểu đường tuýp 2, người ta không thể bỏ qua các chỉ số y tế lâm sàng – vì đây là những dấu hiệu quan trọng để bác sĩ chẩn đoán, và cũng là dữ liệu đầu vào chủ yếu cho các mô hình học máy dự đoán bệnh. Trong đề tài này, em đã sử dụng những chỉ số đơn giản, dễ thu thập, nhưng có giá trị dự báo cao, thường gặp trong hồ sơ khám sức khỏe tổng quát.

Dưới đây là các chỉ số được chọn đưa vào mô hình và lý do vì sao chúng quan trọng:

**Tuổi (Age):** Tuổi càng cao, nguy cơ mắc tiểu đường càng lớn. Điều này liên quan đến quá trình lão hóa, suy giảm chức năng tế bào beta tuyến tụy, tăng kháng insulin và tích lũy mỡ nội tạng. Theo WHO, từ sau 40 tuổi, nguy cơ mắc tiểu đường tăng mạnh, đặc biệt ở người có lối sống ít vận động [42].

**Giới tính (Gender):** Một số nghiên cứu cho thấy nam giới có nguy cơ khởi phát sớm hơn, trong khi phụ nữ sau mãn kinh lại có nguy cơ tăng đột biến do thay

đổi nội tiết. Hơn nữa, thói quen ăn uống và vận động thường khác nhau giữa nam và nữ, ảnh hưởng đến tỷ lệ mắc bệnh.

**Chỉ số đường huyết (Glucose):** Đây là chỉ số quan trọng nhất. Giá trị glucose tăng cao là dấu hiệu đặc trưng của bệnh tiểu đường. Theo ADA (Hiệp hội Tiểu đường Hoa Kỳ):

- Đường huyết lúc đói  $\geq 126$  mg/dL ( $\geq 7.0$  mmol/L): được xem là tiêu chuẩn chẩn đoán tiểu đường.
- Từ 100–125 mg/dL (5.6–6.9 mmol/L): gọi là tiền tiểu đường [7].

**Huyết áp tâm thu & tâm trương (Systolic & Diastolic Blood Pressure):** Tăng huyết áp là yếu tố nguy cơ quan trọng vì nó thường đi kèm với tiểu đường trong hội chứng chuyển hóa.

- Huyết áp tâm thu (systolic): là áp lực khi tim co bóp, bình thường ~120 mmHg.
- Huyết áp tâm trương (diastolic): là áp lực khi tim nghỉ giữa 2 nhịp đập, bình thường ~80 mmHg.

Theo WHO, người có huyết áp  $\geq 140/90$  mmHg có nguy cơ mắc tiểu đường cao gấp đôi so với người bình thường [5, 43].

**Nhịp tim (Pulse rate):** Nhịp tim là chỉ số sinh tồn dễ đo. Nhịp tim cao bất thường có thể là dấu hiệu của stress, rối loạn chuyển hóa hoặc hệ thần kinh tự chủ hoạt động không bình thường – điều thường gặp ở người có nguy cơ tiểu đường. Một nghiên cứu của Đại học Harvard (2020) cho thấy: nhịp tim khi nghỉ cao hơn 80 nhịp/phút có liên quan đến nguy cơ tiểu đường tuýp 2 tăng đáng kể [21].

**Chiều cao, cân nặng và chỉ số BMI:** BMI (Body Mass Index) = Cân nặng (kg) / [Chiều cao (m)]<sup>2</sup>. BMI là cách đơn giản nhất để phân loại mức độ thừa cân, yếu tố nguy cơ hàng đầu gây ra tiểu đường tuýp 2.

Theo WHO:

- BMI từ 25 đến 29.9 → thừa cân.
- BMI  $\geq 30$  → béo phì → nguy cơ tiểu đường tăng mạnh.

**Tiền sử gia đình mắc tiểu đường (Family Diabetes):** Nếu trong gia đình có người mắc tiểu đường, nguy cơ di truyền rất cao (tăng 2–6 lần). Biến này phản ánh yếu tố gen và thói quen ăn uống, lối sống chung trong gia đình [5, 43].

**Các yếu tố liên quan khác:** Ngoài các chỉ số chính, mô hình còn bổ sung các biến y tế quan trọng khác.

- Tăng huyết áp mạn tính (Hypertensive): người đã được chẩn đoán là cao huyết áp.
- Tiền sử đột quỵ (Stroke): vì đột quỵ và tiểu đường thường liên quan nhau.
- Bệnh tim mạch (Cardiovascular disease): mối liên hệ chặt chẽ với tiểu đường.
- Tiền sử gia đình có người bị cao huyết áp (Family Hypertension): phản ánh di truyền tim mạch.

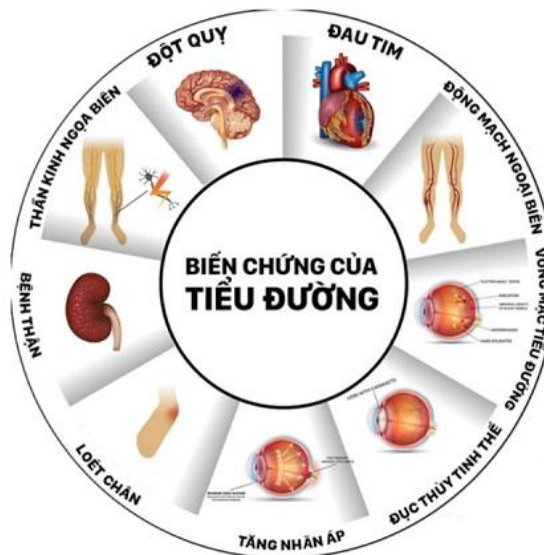
### **2.1.5 Biến chứng của bệnh tiểu đường**

Một trong những điều khiến bệnh tiểu đường trở nên nguy hiểm là diễn biến âm thầm nhưng lại gây hậu quả nghiêm trọng về lâu dài. Nhiều bệnh nhân tiểu đường không cảm thấy gì trong thời gian đầu mắc bệnh, nhưng sau vài năm, khi phát hiện thì đã bị biến chứng ở mắt, thận, tim mạch.

Tiểu đường gây biến chứng theo hai hướng chính:

**Biến chứng cấp tính:** xảy ra nhanh, có thể đe dọa tính mạng nếu không xử lý kịp thời.

**Biến chứng mạn tính:** âm thầm xuất hiện sau nhiều năm, ảnh hưởng nghiêm trọng đến chất lượng sống.



Hình 2. 3. Biến chứng của bệnh tiểu đường

### Biến chứng cấp tính

Hạ đường huyết (hypoglycemia): Xảy ra khi đường huyết tụt quá thấp (thường dưới 3.9 mmol/L hoặc 70 mg/dL), thường do dùng thuốc hạ đường huyết hoặc insulin quá liều. Triệu chứng: run tay, đổ mồ hôi, tim đập nhanh, choáng váng, lú lẫn, thậm chí hôn mê nếu không xử lý kịp thời. Đây là biến chứng thường gặp ở người đã được chẩn đoán và điều trị, nhưng cũng cảnh báo rằng bệnh cần theo dõi kỹ.

Tăng đường huyết cấp (nhiễm toan ceton hoặc hôn mê tăng áp lực thẩm thấu): Trong trường hợp cơ thể không có đủ insulin, mỡ sẽ bị phân giải để tạo năng lượng → sinh ra chất độc ketone. Nếu ketone tích tụ quá nhiều → nhiễm toan máu, nôn ói, thở nhanh, hôn mê, có thể tử vong nếu không cấp cứu kịp. Thường gặp ở người tuýp 1 hoặc người tuýp 2 bỏ thuốc, nhiễm trùng nặng.

Theo WHO, mỗi năm có hàng trăm ngàn ca tử vong do biến chứng cấp của tiểu đường, phần lớn có thể phòng tránh được [43].

### Biến chứng mạn tính (dài hạn):

Biến chứng mạn tính xuất hiện sau nhiều năm mắc bệnh, đặc biệt khi người bệnh không kiểm soát đường huyết tốt. Đây là nguyên nhân chính gây tàn tật, tử vong sớm và suy giảm chất lượng sống ở bệnh nhân tiểu đường.

Biến chứng tim mạch: Là nguyên nhân tử vong hàng đầu ở người tiểu đường. Bệnh nhân có nguy cơ đột quỵ, nhồi máu cơ tim, xơ vữa động mạch cao gấp 2–4 lần người bình thường [43]. Tăng đường huyết làm tổn thương thành mạch máu, kết hợp với mỡ máu cao và huyết áp cao tạo thành “tam giác nguy hiểm”.

Biến chứng ở thận (bệnh thận do tiểu đường): Tiểu đường làm tổn thương các mao mạch nhỏ trong cầu thận → lọc máu kém → đạm xuất hiện trong nước tiểu (tiểu albumin). Nếu không phát hiện và kiểm soát sớm, người bệnh có thể suy thận mạn, phải chạy thận nhân tạo suốt đời.

Theo ADA (2023), khoảng 30–40% bệnh nhân tiểu đường sẽ bị tổn thương thận ở mức độ nào đó trong suốt quá trình mắc bệnh [7].

Biến chứng ở mắt: Tổn thương võng mạc (diabetic retinopathy) là một trong những nguyên nhân gây mù lòa hàng đầu ở người trưởng thành. Ngoài ra còn có thể bị: đục thủy tinh thể, tăng nhãn áp. Đặc biệt nguy hiểm vì người bệnh không thấy mờ ngay từ đầu, mà chỉ phát hiện khi thị lực suy giảm nặng.

Tổn thương thần kinh (neuropathy): Người bệnh có thể bị: tê, nóng rát, mất cảm giác, co thắt cơ. Tổn thương thường bắt đầu từ bàn chân → nếu không để ý, có thể bị nhiễm trùng nặng hoặc hoại tử. Đây là lý do vì sao có nhiều bệnh nhân tiểu đường bị cắt cụt ngón chân, bàn chân.

Biến chứng trên hệ miễn dịch và nhiễm trùng: Người tiểu đường dễ bị nhiễm trùng hơn do hệ miễn dịch hoạt động yếu. Vết thương lâu lành, dễ bị viêm mô tế bào, viêm tiết niệu, nhiễm nấm da. Nếu không kiểm soát tốt đường huyết, các nhiễm trùng có thể diễn biến rất nhanh và khó điều trị.

#### **2.1.6 Tác động xã hội và chi phí y tế của bệnh tiểu đường**

Tiểu đường không chỉ là một bệnh lý đơn lẻ, mà nó còn là một gánh nặng y tế, kinh tế và xã hội đang gia tăng ở hầu hết các quốc gia, đặc biệt là ở các nước đang phát triển như Việt Nam. Sự ảnh hưởng của căn bệnh này không dừng lại ở người mắc, mà còn lan ra cả gia đình, cộng đồng và hệ thống y tế quốc gia.

Vì vậy, việc đánh giá đúng mức độ ảnh hưởng của tiểu đường là rất quan trọng trong chiến lược dự phòng và kiểm soát.



Gánh nặng tài chính đối với người bệnh: Một trong những điều khiến người mắc tiểu đường lo lắng nhất là chi phí điều trị kéo dài và tốn kém. Khác với bệnh cấp tính như cảm cúm, tiểu đường không thể chữa khỏi hoàn toàn, mà cần điều trị suốt đời.

Người bệnh phải đi khám định kỳ, xét nghiệm máu, kiểm tra mắt, thận, tim. Mỗi tháng có thể tốn hàng trăm nghìn đến vài triệu đồng cho thuốc hạ đường huyết, insulin, máy đo đường huyết, test que. Nếu có biến chứng như suy thận, mù lòa hoặc cắt cụt chi, chi phí điều trị sẽ tăng gấp nhiều lần.

Theo báo cáo của Bộ Y tế Việt Nam (2021), chi phí trung bình điều trị cho một bệnh nhân tiểu đường có biến chứng có thể lên tới 20–40 triệu đồng/năm – con số vượt quá khả năng chi trả của nhiều hộ gia đình [4].

Giảm năng suất lao động và thu nhập: Tiểu đường thường đi kèm với mệt mỏi, kiệt sức, làm giảm khả năng làm việc. Nhiều người bệnh phải nghỉ làm thường xuyên để đi tái khám, nhập viện hoặc chạy thận. Nếu người bệnh là lao động chính trong gia đình thì tác động sẽ càng nghiêm trọng.

Tăng áp lực cho hệ thống y tế: Số ca tiểu đường ngày càng tăng nhanh khiến các bệnh viện tuyến tỉnh, tuyến trung ương luôn trong tình trạng quá tải, đặc biệt ở khoa nội tiết, thận nhân tạo, tim mạch. Điều trị biến chứng (suy thận, loét chân, đột quỵ...) tốn kém hơn rất nhiều so với điều trị tiểu đường giai đoạn đầu. Nếu không có biện pháp dự phòng hiệu quả, chi phí y tế quốc gia sẽ ngày càng tăng, ảnh hưởng đến ngân sách bảo hiểm y tế.

Theo Liên đoàn Đái tháo đường Thế giới (IDF, 2021), tổng chi phí y tế trực tiếp cho bệnh tiểu đường toàn cầu đã vượt 966 tỷ USD vào năm 2021 – tăng gần gấp đôi chỉ trong 15 năm [22].

Tác động tâm lý – xã hội lâu dài: Bệnh nhân tiểu đường thường phải sống với tâm lý lo âu, trầm cảm nhẹ, vì sợ biến chứng, phải kiêng khem và chịu nhiều hạn chế trong sinh hoạt. Nhiều người cảm thấy “không còn là chính mình”, ngại ra ngoài, ngại giao tiếp. Đối với trẻ em mắc tiểu đường tuýp 1, áp lực tâm lý còn lớn hơn vì phải mang theo insulin, máy đo, và kiêng khem từ nhỏ. Ngoài ra, còn có ảnh

hưởng về giới tính, sinh sản, đặc biệt ở phụ nữ có thai bị tiểu đường thai kỳ, dễ bị kỳ thị hoặc lo lắng quá mức.

Tác động đến cộng đồng và quốc gia: Tiểu đường được xem là một trong những nguyên nhân hàng đầu gây tử vong sớm không do lây nhiễm, cùng với tim mạch và ung thư. Khi bệnh không được kiểm soát tốt thì tỉ lệ tử vong sớm tăng, giảm tuổi thọ bình quân của quốc gia. Người dân mắc bệnh nhiều sẽ là gánh nặng cho quỹ bảo hiểm y tế và ngân sách quốc gia.

Theo WHO (2023), bệnh tiểu đường gây tử vong cho khoảng 6,7 triệu người mỗi năm, tức cứ 5 giây có 1 người chết vì tiểu đường hoặc các biến chứng liên quan [43].

### **2.1.7 Vai trò của công nghệ trong phòng ngừa bệnh tiểu đường**

Trong thời đại công nghệ phát triển mạnh mẽ như hiện nay, các ứng dụng công nghệ ngày càng đóng vai trò quan trọng trong việc phát hiện sớm, phòng ngừa và quản lý bệnh tiểu đường. Thay vì chỉ dựa vào chẩn đoán khi bệnh đã xảy ra, công nghệ đang giúp con người chủ động hơn trong việc theo dõi sức khỏe và ngăn ngừa nguy cơ từ sớm – điều đặc biệt quan trọng đối với tiểu đường tuýp 2.

Thiết bị đo lường và theo dõi sức khỏe: Ngày nay, người dân có thể dễ dàng tiếp cận với nhiều thiết bị y tế cá nhân. Máy đo đường huyết tại nhà giúp kiểm tra đường máu hằng ngày mà không cần đến bệnh viện. Máy đo huyết áp, cân điện tử, thiết bị đo nhịp tim: hỗ trợ theo dõi các chỉ số liên quan đến nguy cơ chuyển hóa. Smartwatch là vòng tay sức khỏe (ví dụ Fitbit, Apple Watch): đo bước chân, nhịp tim, theo dõi giấc ngủ – từ đó đánh giá lối sống và hoạt động thể chất.

Ứng dụng y tế trên điện thoại và máy tính: Hiện nay có rất nhiều ứng dụng được thiết kế chuyên biệt cho người có nguy cơ hoặc đang sống chung với bệnh tiểu đường. MySugr, BlueStar Diabetes, Glucose Buddy là các ứng dụng cho phép người dùng ghi lại đường huyết, thức ăn, thuốc và vận động.

Một số ứng dụng còn có AI hỗ trợ đánh giá xu hướng và đưa ra cảnh báo, nhắc nhở dùng thuốc, gợi ý bữa ăn. Tuy nhiên, đa phần các ứng dụng này chỉ tập trung vào người đã mắc bệnh, còn việc dự đoán nguy cơ mắc bệnh từ sớm vẫn chưa được phổ cập rộng rãi, đặc biệt ở các nước đang phát triển như Việt Nam.

Trí tuệ nhân tạo (AI) và học máy (Machine Learning): Trí tuệ nhân tạo – đặc biệt là học máy – đang trở thành một hướng đi mới trong phòng ngừa bệnh tiểu đường. AI có thể học từ hàng nghìn dữ liệu bệnh nhân, tìm ra mô hình nguy cơ mà mắt người không thể thấy được.

Các mô hình như Logistic Regression, Random Forest, Neural Network có thể dự đoán khả năng mắc bệnh dựa trên các chỉ số như: tuổi, BMI, đường huyết, huyết áp, tiền sử. Việc áp dụng mô hình học máy vào hệ thống ứng dụng web hoặc app điện thoại sẽ giúp người dùng tự kiểm tra nguy cơ bệnh tại nhà, ngay cả khi chưa có triệu chứng.

Theo báo cáo của The Lancet Digital Health (2021), các mô hình học máy đã đạt độ chính xác từ 80–90% trong việc dự đoán tiểu đường khi được huấn luyện đúng cách và sử dụng dữ liệu phù hợp [39].

Tăng cường nhận thức và thay đổi hành vi: Một vai trò rất thiết thực của công nghệ là giúp người dân nhận ra mình có nguy cơ và thay đổi hành vi sớm hơn. Ứng dụng gợi ý bài tập phù hợp, lượng calo cần đốt, nhắc nhở uống nước, ngủ đủ giấc, ăn đúng giờ, Một số hệ thống còn đánh giá nguy cơ theo thói quen ăn uống được ghi lại (diet tracking).

Theo WHO (2020), nếu người có tiền sử gia đình mắc tiểu đường thay đổi lối sống kịp thời, nguy cơ chuyển thành bệnh thực sự có thể giảm tới 58% [41].

Thực trạng ở Việt Nam – Cơ hội cho giải pháp đơn giản, dễ tiếp cận: Tại Việt Nam, đa số người dân chưa có thói quen kiểm tra sức khỏe định kỳ hoặc không đủ điều kiện để tiếp cận các công cụ chẩn đoán chuyên sâu.

Việc xây dựng một mô hình cảnh báo sớm dựa trên các chỉ số y tế cơ bản như: tuổi, đường huyết, BMI, huyết áp, tiền sử là rất thực tế và phù hợp. Nếu kết hợp được mô hình này vào một ứng dụng web miễn phí, giao diện đơn giản, tiếng Việt, thì hoàn toàn có thể triển khai rộng trong cộng đồng.

## **2.2 Một số công trình và sản phẩm tương tự**

Để thực hiện tốt đề tài, em đã tham khảo một số công trình nghiên cứu và sản phẩm liên quan đến việc ứng dụng học máy trong dự đoán bệnh tiểu đường. Qua quá trình tìm hiểu, em nhận thấy rằng mặc dù trên thế giới đã có nhiều nghiên

cứu sâu rộng, nhưng đa phần vẫn tập trung vào các tập dữ liệu và bối cảnh y tế tại châu Âu hoặc Bắc Mỹ.

Trong khi đó, việc áp dụng tại khu vực châu Á, đặc biệt là môi trường giống Việt Nam, còn khá hạn chế. Vì vậy, đề tài này hướng tới mục tiêu tận dụng mô hình đơn giản, dữ liệu gần thực tế, nhưng vẫn hiệu quả và dễ triển khai.

### **2.2.1 Các nghiên cứu và mô hình học thuật nổi bật**

Nghiên cứu sử dụng tập dữ liệu PIMA Indian Diabetes Dataset (PID): PIMA là một trong những tập dữ liệu y tế kinh điển được dùng để huấn luyện và kiểm tra các mô hình học máy trong bài toán phân loại tiểu đường. Tập dữ liệu này được công bố bởi National Institute of Diabetes and Digestive and Kidney Diseases (NIDDK), thu thập từ phụ nữ người Mỹ gốc da đỏ (Pima Indian).

Gồm 768 dòng dữ liệu và 8 đặc trưng đầu vào: số lần mang thai, glucose, huyết áp, độ dày da, insulin, BMI, tuổi và chỉ số di truyền. Nhiều nghiên cứu đã áp dụng các mô hình như: Logistic Regression, K-Nearest Neighbors, SVM, Decision Tree, Random Forest, thậm chí Deep Learning để dự đoán tiểu đường từ tập dữ liệu này. Tỷ lệ nhãn dương/lệch khá lớn (tầm 65% âm tính, 35% dương tính), giúp kiểm chứng khả năng phân loại của thuật toán.

Các phương pháp học máy phổ biến như Logistic Regression, Random Forest và SVM được xem là nền tảng trong nhiều nghiên cứu dự đoán bệnh tiểu đường [23].

Theo nghiên cứu của Smith et al. (2020), Logistic Regression trên tập PIMA đạt độ chính xác 76%, trong khi Random Forest có thể đạt 82% [36].

Ứng dụng mô hình học máy trong phân tích y tế toàn cầu: Theo The Lancet Digital Health (2021), nhiều quốc gia tiên tiến đã bắt đầu tích hợp AI vào hệ thống y tế cộng đồng. Sử dụng các mô hình hồi quy (Logistic Regression), cây quyết định và mạng nơ-ron sâu để phân tích hồ sơ bệnh án điện tử (EHR) và dự đoán nguy cơ tiểu đường sớm. Một số mô hình kết hợp thêm dữ liệu hình ảnh (retina, MRI...) cho bài toán chẩn đoán biến chứng mắt do tiểu đường [39]. Tuy nhiên, đa số các hệ thống này yêu cầu hạ tầng mạnh, dữ liệu đầy đủ và khó tiếp cận với cộng đồng bình thường.

Nghiên cứu tại châu Á – Bangladesh Dataset: Gần đây, một số nhóm nghiên cứu đã hướng đến việc xây dựng mô hình học máy trên dữ liệu thực tế tại các nước đang phát triển, tiêu biểu là Bangladeshi Dataset for Type 2 Diabetes, do nhóm chuyên gia y tế ở Dhaka thu thập. Bộ dữ liệu gồm ~5200 mẫu, 14 đặc trưng, rất sát với điều kiện dân số và sức khỏe của Việt Nam. Các biến như: glucose, BMI, huyết áp, nhịp tim, tiền sử gia đình... được đo lường lâm sàng và chuẩn hóa tốt. Trong nhiều nghiên cứu, Logistic Regression tỏ ra rất hiệu quả trên bộ dữ liệu này, nhờ khả năng diễn giải rõ ràng và không cần huấn luyện phức tạp.

Nghiên cứu của Faruque et al. (2021) sử dụng Logistic Regression trên Bangladeshi Dataset đạt độ chính xác 84,7%, vượt qua cả SVM và KNN khi xử lý bằng dữ liệu đã chuẩn hóa [15].

### 2.2.2 Khảo sát các sản phẩm y tế ứng dụng thực tế

Bên cạnh các nghiên cứu học thuật, em cũng tìm hiểu và khảo sát một số ứng dụng thực tế đã và đang được triển khai để hỗ trợ người mắc bệnh tiểu đường. Những sản phẩm này phần lớn đều đã được thương mại hóa hoặc đưa vào sử dụng tại các quốc gia phát triển, chủ yếu tập trung vào việc theo dõi chỉ số sức khỏe, hỗ trợ điều trị, tư vấn chế độ ăn uống cho người đã mắc bệnh. Tuy nhiên, đa phần chưa chú trọng đến cảnh báo nguy cơ sớm cho người chưa mắc bệnh, đặc biệt ở các nước có thu nhập trung bình như Việt Nam.

**MySugr (Áo):** Là một trong những ứng dụng theo dõi tiểu đường được dùng phổ biến nhất tại châu Âu và Mỹ. Cho phép người bệnh ghi lại đường huyết, lượng carbohydrate, insulin, tập thể dục, tâm trạng. Giao diện đơn giản, có các biểu đồ và gợi ý dựa trên dữ liệu đầu vào. Có bản miễn phí và bản trả phí (đầy đủ hơn) [27].

Ưu điểm:

- Tương thích với nhiều máy đo đường huyết.
- Dễ sử dụng, tích hợp AI gợi ý nhắc nhở.

Nhược điểm:

- Phù hợp hơn với người đã được chẩn đoán bệnh.

- Ngôn ngữ chủ yếu là ngôn ngữ tiếng Anh và tiếng Đức.

**BlueStar Diabetes (Hoa Kỳ):** Là ứng dụng được FDA (Cục Quản lý Dược Hoa Kỳ) phê duyệt, hỗ trợ quản lý tiểu đường theo đúng phác đồ y tế. Có khả năng kết nối với bác sĩ điều trị, gợi ý tự động chế độ ăn và vận động hàng ngày. Gửi cảnh báo nếu người dùng quên dùng thuốc hoặc ghi nhận chỉ số bất thường [14].

Ưu điểm:

- Có tính pháp lý cao, chuyên biệt hóa theo phác đồ điều trị.
- Phù hợp với bệnh nhân có bảo hiểm và theo dõi lâu dài.

Nhược điểm:

- Yêu cầu tài khoản đăng ký y tế tại Mỹ.
- Cài đặt và vận hành phức tạp hơn với người lớn tuổi.

**Glucose Buddy (Mỹ):** Là một ứng dụng nổi bật với giao diện thân thiện, hỗ trợ người dùng theo dõi chỉ số đường huyết, insulin, thuốc, bước đi, bữa ăn. Tích hợp dữ liệu từ Apple Health và Google Fit [17].

Ưu điểm:

- Dễ dùng với người dùng phổ thông.
- Có chức năng “tự học” theo thói quen người dùng.

Nhược điểm:

- Tập trung vào quản lý bệnh hơn là dự đoán sớm nguy cơ.
- Không có ngôn ngữ tiếng Việt.

**Apple Health:** Đây là hệ sinh thái sức khỏe tích hợp sẵn trên điện thoại thông minh iOS. Không chuyên biệt cho tiểu đường, nhưng giúp theo dõi vận động, nhịp tim, ngủ, dinh dưỡng. Dữ liệu từ đồng hồ thông minh, cân điện tử, máy đo huyết áp được đồng bộ trực tiếp [8].

Ưu điểm:

- Miễn phí, dễ tiếp cận.
- Đã có mặt ở Việt Nam và được nhiều người trẻ sử dụng.

Nhược điểm:

- Không đưa ra cảnh báo tiểu đường hoặc phân tích nguy cơ.
- Không chuyên sâu về bệnh lý.

### **2.2.3 Hạn chế của các nghiên cứu và sản phẩm hiện tại**

Sau khi tìm hiểu các nghiên cứu học thuật (mục 2.2.1) và khảo sát một số ứng dụng thực tế đang được sử dụng phổ biến (mục 2.2.2), em nhận thấy rằng vẫn còn khá nhiều khoảng trống chưa được giải quyết triệt để. Đây cũng chính là lý do mà đề tài này được chọn nhằm khắc phục một phần những hạn chế đó và đóng góp một giải pháp đơn giản nhưng hiệu quả cho cộng đồng.

Thiếu giải pháp cho giai đoạn trước khi mắc bệnh: Hầu hết các nghiên cứu và sản phẩm hiện tại tập trung vào người đã được chẩn đoán mắc tiểu đường, chứ chưa hướng mạnh vào việc dự đoán nguy cơ mắc bệnh từ trước – tức là giai đoạn tiền tiểu đường.

Trong khi đó, tiền tiểu đường chiếm tới 30–40% dân số trưởng thành tại nhiều nước, và rất nhiều người không biết mình đang ở nhóm nguy cơ cao. Nếu phát hiện sớm trong giai đoạn này, thay đổi lối sống đơn giản như ăn uống lành mạnh, tập thể dục... có thể giúp phòng bệnh hoàn toàn.

WHO (2020) nhấn mạnh: “80% bệnh tiểu đường tuýp 2 có thể phòng ngừa được nếu phát hiện sớm các yếu tố nguy cơ” [41].

Hạn chế về khả năng tiếp cận đại chúng: Nhiều ứng dụng hiện có chỉ phù hợp với người có điều kiện. Đa phần là ngôn ngữ tiếng Anh hoặc ngôn ngữ châu Âu, chưa phổ biến ở Việt Nam. Cần máy đo đường huyết, cảm biến, kết nối với đồng hồ thông minh, không phải người nào cũng có hoặc biết sử dụng. Một số ứng dụng yêu cầu tài khoản y tế quốc tế, hoặc chi phí sử dụng khá cao (bản premium, thuê chuyên gia).

Dữ liệu huấn luyện chưa phù hợp với đặc điểm dân số Việt Nam: Phần lớn các mô hình học máy hiện tại được huấn luyện trên tập dữ liệu PIMA (người da đỏ Mỹ), dữ liệu bệnh án điện tử từ châu Âu, Mỹ, vốn khác biệt khá lớn với thể trạng người châu Á, thói quen ăn uống, điều kiện kinh tế, môi trường sống. Điều này

khiến mô hình đôi khi hoạt động không chính xác khi áp dụng cho người Việt Nam, dẫn đến hiểu nhầm hoặc đánh giá sai mức độ nguy cơ.

Quá phức tạp trong sử dụng: Không ít ứng dụng hoặc mô hình nghiên cứu chỉ phù hợp với người có kiến thức chuyên môn, biết đọc kết quả thống kê, biểu đồ chuyên sâu, biết cách nhập dữ liệu đúng định dạng. Với người dùng phổ thông, nhập đơn giản các thông tin như tuổi, chiều cao, cân nặng, đường huyết, huyết áp. Sau đó nhận kết quả rõ ràng, ví dụ: “Nguy cơ thấp – trung bình – cao và xác suất bao nhiêu %”. Đây là điều mà nhiều hệ thống học máy hiện tại chưa tối ưu được về giao diện và trải nghiệm người dùng.

Chưa tích hợp khả năng giáo dục sức khỏe cộng đồng: Một số sản phẩm rất tốt về mặt công nghệ, nhưng vẫn còn thiếu các yếu tố như không có phân tích giải thích vì sao người đó có nguy cơ, không gợi ý thay đổi hành vi hoặc theo dõi sức khỏe theo thời gian và không tích hợp chức năng lưu lịch sử dự đoán.

#### **2.2.4 Định hướng lý thuyết từ khảo sát**

Từ những khảo sát thực tế và nghiên cứu đã trình bày ở các mục trên, em nhận thấy rằng để xây dựng một mô hình dự đoán nguy cơ tiểu đường thực tế, hiệu quả và dễ tiếp cận, cần phải xác định rõ những nguyên tắc lý thuyết cốt lõi. Đây chính là nền tảng để em định hướng toàn bộ cấu trúc đề tài – từ cách chọn dữ liệu, lựa chọn thuật toán học máy, đến thiết kế hệ thống giao diện.

Lựa chọn mô hình đơn giản nhưng có khả năng giải thích rõ ràng: Qua khảo sát, em thấy nhiều mô hình học máy hiện đại như Deep Learning có độ chính xác rất cao, nhưng lại khó triển khai ở thực tế do cần dữ liệu lớn, hạ tầng mạnh. Không thể giải thích vì sao dự đoán nguy cơ cao hay thấp, không thân thiện với người không chuyên.

Do đó, em ưu tiên hướng đến các mô hình dễ triển khai, có khả năng diễn giải tốt, tiêu biểu là:

Logistic Regression: cho phép phân tích ảnh hưởng của từng biến đầu vào tới khả năng mắc bệnh, đồng thời kết quả là xác suất rõ ràng.

Random Forest hoặc Decision Tree: hỗ trợ nếu muốn tăng độ chính xác, nhưng vẫn giữ được tính dễ hiểu.



Điều này phù hợp với nhận định từ ADA (2023): “Các mô hình đơn giản nhưng có thể giải thích được vẫn là lựa chọn ưu tiên trong hệ thống y tế cộng đồng” [7].

Chọn đặc trưng đầu vào gần gũi, dễ thu thập: Qua khảo sát các bộ dữ liệu như PIMA, Bangladeshi, em thấy các đặc trưng như tuổi, giới tính, BMI, huyết áp, đường huyết, tiền sử gia đình... là những chỉ số ai cũng có thể tự khai báo hoặc đo bằng thiết bị đơn giản tại nhà hoặc trạm y tế cơ sở.

Vì vậy, việc lựa chọn các chỉ số y tế không đòi hỏi xét nghiệm chuyên sâu sẽ giúp mô hình tiếp cận được với nhiều người hơn – đặc biệt ở vùng nông thôn, người thu nhập thấp hoặc người không rành công nghệ.

Giao diện và hệ thống cần đơn giản, trực quan: Từ khảo sát các ứng dụng thực tế (mục 2.2.2), em rút ra một số nguyên tắc về giao diện. Giao diện ngôn ngữ tiếng Việt, đơn giản, dễ dùng cho mọi đối tượng, cần tài khoản để lưu lịch sử dự đoán.

Người dùng chỉ cần nhập thông tin như tuổi, chỉ số đường huyết, huyết áp, bmi thì hệ thống sẽ đưa ra kết quả phân loại nguy cơ thấp – trung bình – cao. Điều này giúp tạo trải nghiệm thân thiện, giúp người dùng cảm thấy dễ hiểu, gần gũi và không bị quá chuyên sâu về học thuật.

Tập dữ liệu chọn lọc, sát thực tế Việt Nam: Thay vì dùng tập PIMA cổ điển, em lựa chọn Bangladeshi Dataset for Type 2 Diabetes vì có cấu trúc gần gũi với đặc điểm dân số, y tế, chỉ số sinh học của người Việt.

Đầy đủ các biến thuộc tính đặc trưng như tuổi, chỉ số đường huyết (glucose), BMI, huyết áp, nhịp tim, tiền sử gia đình. Việc lựa chọn tập dữ liệu phù hợp không chỉ giúp tăng độ chính xác của mô hình, mà còn giúp tăng giá trị ứng dụng thực tế, đây là điều em đặc biệt quan tâm trong khóa luận này.

Ưu tiên khả năng triển khai nhanh, chi phí thấp, dễ nhân rộng: Từ những hạn chế của các sản phẩm đang có (mục 2.2.3), em hướng đến hệ thống ứng dụng website thay vì app mobile sẽ giúp tiết kiệm chi phí phát triển, dễ cập nhật, không cần cài đặt.

Sử dụng mô hình học máy nhẹ, dễ triển khai với Flask và ReactJS. Hệ thống có thể được triển khai thử nghiệm ở các trường học, trung tâm y tế chỉ cần máy tính và kết nối mạng.

### **2.2.5 Tính mới trong cách tiếp cận lý thuyết**

Sau khi khảo sát các hướng nghiên cứu và ứng dụng thực tế, em nhận thấy đề tài này mang lại một số điểm mới mẻ và khác biệt, phù hợp với nhu cầu của xã hội hiện tại, đồng thời có khả năng triển khai thực tiễn cao nếu tiếp tục được phát triển. Những điểm mới này đến từ cả phía dữ liệu, mô hình học máy, cách xây dựng hệ thống, lẫn định hướng cộng đồng.

Tập dữ liệu gần với thể trạng người Việt: Đề tài sử dụng Bangladeshi Dataset for Type 2 Diabetes – một bộ dữ liệu có cấu trúc gần gũi với đặc điểm sinh lý, dinh dưỡng, và điều kiện sống của người Việt Nam hơn so với các tập dữ liệu cổ điển như PIMA. Dữ liệu gồm hơn 5000 mẫu, có sẵn các đặc trưng phổ biến như: glucose, BMI, huyết áp, tuổi, giới tính, tiền sử. Tỷ lệ nhãn phân bố đều, dễ xử lý và huấn luyện mô hình.

Đây là một bước cải tiến giúp mô hình tăng độ chính xác khi triển khai trong cộng đồng người Việt. Xây dựng giao diện web thân thiện hoàn toàn bằng tiếng Việt, dễ dùng với mọi đối tượng, kể cả người không rành công nghệ hay người lớn tuổi.

Tập trung vào giai đoạn tiền tiểu đường (pre-diabetes): Đa số các hệ thống hiện nay tập trung vào hỗ trợ người đã mắc bệnh, trong khi đề tài này tập trung vào cảnh báo nguy cơ từ sớm, ngay cả khi người dùng chưa có triệu chứng. Đây là hướng tiếp cận mang tính phòng ngừa, góp phần giảm gánh nặng y tế nếu được ứng dụng rộng rãi. Dễ triển khai tại cộng đồng, trường học, công sở, nơi mà các chương trình chưa phổ biến.

Mô hình đơn giản, hiệu quả, có khả năng diễn giải: Thay vì chạy theo các mô hình phức tạp như deep learning, đề tài chọn Logistic Regression là một mô hình đơn giản nhưng hiệu quả và giải thích được. Kết quả mô hình là xác suất mắc bệnh dễ hiểu với người không chuyên. Có thể biết rõ từng yếu tố như tuổi, BMI,

huyết áp ảnh hưởng ra sao đến kết quả. Giúp mô hình trở nên mở và giải thích được điều rất quan trọng khi triển khai trong thực tế.

### **2.2.6 Khả năng ứng dụng thực tiễn**

Một trong những mục tiêu quan trọng nhất của đề tài không chỉ là đạt được kết quả dự đoán tốt trong môi trường thử nghiệm, mà còn là có thể triển khai và ứng dụng rộng rãi trong thực tế cuộc sống.

Sau quá trình nghiên cứu, em nhận thấy rằng hệ thống được xây dựng trong đề tài này có tiềm năng triển khai cao, đặc biệt là trong các bối cảnh cộng đồng, trường học, doanh nghiệp, và trạm y tế cơ sở

Hình thức triển khai đơn giản, không yêu cầu hạ tầng cao: Một máy chủ chạy Flask (backend Python), giao diện ReactJS đơn giản (frontend), không cần cài đặt phần mềm hay app, chỉ cần trình duyệt và Internet là sử dụng được.

Dữ liệu đầu vào dễ thu thập: Người dùng chỉ cần nhập các thông tin cơ bản và phổ biến như tuổi, giới tính, cân nặng, chiều cao, chỉ số đường huyết, huyết áp, nhịp tim, tiền sử gia đình. Các chỉ số này có thể được đo tại trạm y tế xã, phòng khám tư, hoặc các thiết bị cá nhân như máy đo huyết áp, máy đo đường huyết tại nhà. Không yêu cầu xét nghiệm chuyên sâu hay kiến thức y khoa phức tạp.

Không yêu cầu người dùng có chuyên môn y tế hoặc công nghệ: Giao diện website được thiết kế thân thiện, bằng tiếng Việt, bố cục rõ ràng. Mọi thao tác chỉ cần click và điền số liệu cơ bản, kết quả được hiển thị dạng biểu đồ, thông báo nguy cơ. Phù hợp với người cao tuổi, người dân vùng nông thôn, học sinh – sinh viên hoặc cán bộ không chuyên ngành y.

Có thể tích hợp vào chương trình truyền thông sức khỏe: Hệ thống có thể được triển khai thí điểm tại các trường học (kết hợp khám sức khỏe định kỳ), các doanh nghiệp (chăm sóc sức khỏe nhân viên), trạm y tế xã/phường (hỗ trợ kiểm soát nguy cơ), các chiến dịch truyền thông sức khỏe cộng đồng do đoàn thể hoặc chính quyền tổ chức giúp tăng hiệu quả tuyên truyền phòng chống bệnh không lây nhiễm, đặc biệt là tiểu đường tuýp 2.

Dễ nâng cấp và tích hợp trong tương lai: Hệ thống có thể lưu trữ lịch sử các chỉ số mà người dùng đã nhập kèm biểu đồ để có thể so sánh lại.

## 2.3 Cơ sở lý thuyết về học máy trong dự đoán bệnh tiểu đường

Trong những năm gần đây, trí tuệ nhân tạo (AI) và học máy (Machine Learning) ngày càng được ứng dụng rộng rãi trong lĩnh vực y tế, từ chẩn đoán bệnh, hỗ trợ ra quyết định lâm sàng, đến cá nhân hóa điều trị.

Đặc biệt, với những bệnh mạn tính phổ biến như tiểu đường tuýp 2, học máy mang lại một công cụ hiệu quả giúp phát hiện sớm nguy cơ, từ đó giúp phòng ngừa biến chứng và giảm gánh nặng y tế lâu dài. Đề tài này sử dụng một mô hình học máy cơ bản – Logistic Regression – để dự đoán khả năng mắc tiểu đường dựa trên các chỉ số y tế cơ bản của người dùng.

### 2.3.1 Khái niệm về học máy (Machine Learning)



Hình 2. 4. Machine Learning – Học máy

Học máy là một nhánh của trí tuệ nhân tạo, nơi mà máy tính có thể học từ dữ liệu để đưa ra dự đoán hoặc quyết định mà không cần được lập trình rõ ràng từng bước.

Hiểu một cách đơn giản, học máy là quá trình mô hình tìm ra mối quan hệ giữa dữ liệu đầu vào (input) và kết quả đầu ra (output) thông qua một tập hợp các ví dụ có sẵn. Khi được cung cấp dữ liệu mới, mô hình sẽ sử dụng kinh nghiệm đã học để dự đoán kết quả tương ứng.

Theo Géron (2019), học máy là một hệ thống có khả năng cải thiện hiệu suất dựa trên kinh nghiệm thu được từ dữ liệu [18].

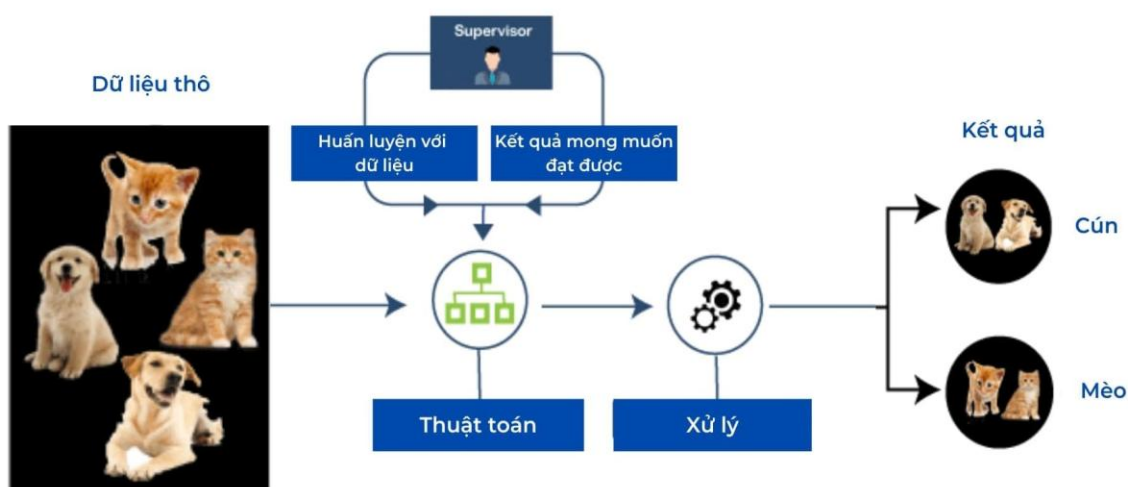
### 2.3.2 Phân loại các mô hình học máy

Trong lĩnh vực học máy, các mô hình được phân loại dựa trên cách mà chúng học từ dữ liệu. Mỗi loại hình sẽ phù hợp với những bài toán và mục tiêu

khác nhau. Theo Géron (2019), học máy được chia làm ba loại chính: học có giám sát, học không giám sát và học tăng cường [18].

Học có giám sát (Supervised Learning): Đây là dạng học phổ biến nhất và được sử dụng trong đề tài này. Trong học có giám sát, dữ liệu đầu vào đi kèm với nhãn (label), tức là đã biết trước kết quả mong muốn. Mô hình sẽ học từ dữ liệu có nhãn để tạo ra một hàm ánh xạ từ đầu vào đến đầu ra.

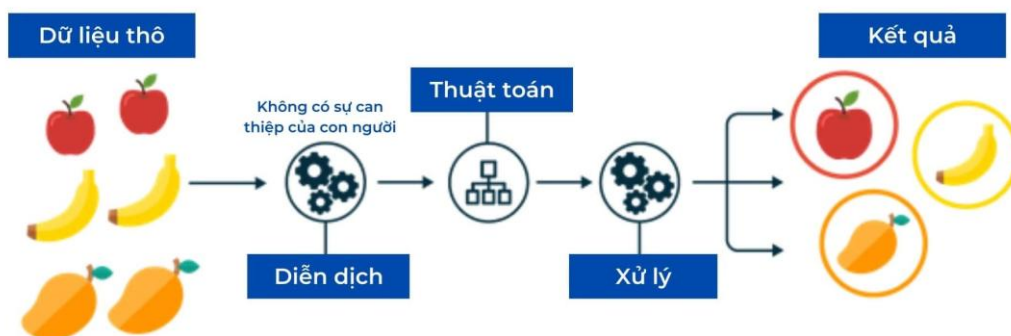
## Học máy có giám sát



Hình 2. 5. Mô hình Học máy có giám sát

Ví dụ, trong bài toán dự đoán tiểu đường, đầu vào gồm các đặc trưng như tuổi, glucose, BMI, huyết áp, nhịp tim. và đầu ra là nhãn 1 (có tiểu đường) hoặc 0 (không có).

## Học máy không giám sát



Hình 2. 6. Mô hình Học máy không giám sát

Học không giám sát (Unsupervised Learning): Trong học không giám sát, dữ liệu không có nhãn. Mục tiêu của mô hình là tìm ra các cấu trúc ẩn, phân nhóm hoặc mối quan hệ trong dữ liệu. Học không giám sát thường được áp dụng trong phân cụm khách hàng, phát hiện bất thường, hoặc giảm chiều dữ liệu.

Học tăng cường (Reinforcement Learning): Là phương pháp học dựa trên tương tác với môi trường. Mô hình nhận được phần thưởng hoặc hình phạt sau mỗi hành động và dần điều chỉnh hành vi để tối ưu hóa kết quả. Học tăng cường phù hợp với game AI, robot hoặc tối ưu chiến lược, không áp dụng trong đề tài này.

### **2.3.3 Bài toán phân loại nhị phân**

Bài toán phân loại là một dạng bài toán học có giám sát, trong đó đầu ra là một hoặc nhiều nhãn rời rạc. Phân loại nhị phân là trường hợp đơn giản nhất – đầu ra chỉ có hai giá trị: ví dụ “Có bệnh” và “Không có bệnh”.

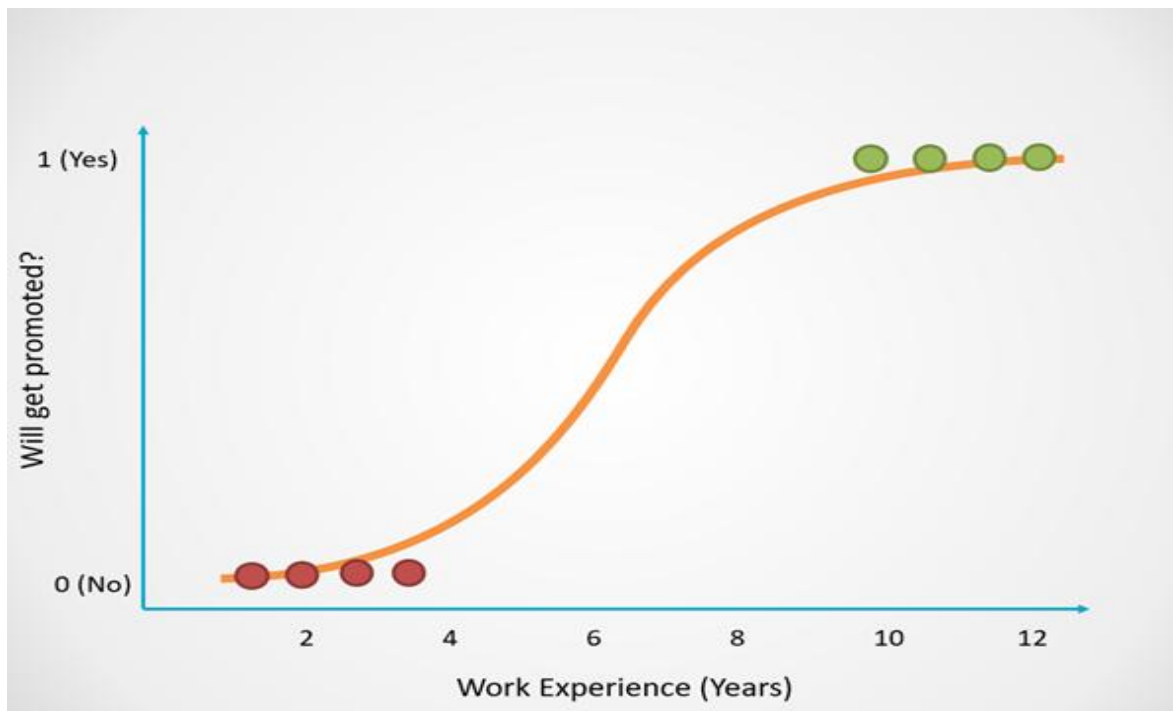
Trong đề tài này, bài toán là phân loại một bệnh nhân thuộc nhóm có nguy cơ mắc bệnh tiểu đường tuýp 2 hay không, dựa trên các đặc trưng như tuổi, chỉ số đường huyết, huyết áp, BMI, nhịp tim, tiền sử gia đình...

Dữ liệu sau khi huấn luyện mô hình sẽ được đưa vào kiểm tra với dữ liệu mới chưa biết kết quả. Kết quả đầu ra là xác suất, và tùy theo ngưỡng định trước (ví dụ 0.5), mô hình sẽ phân loại vào nhóm có hoặc không có nguy cơ.

### **2.3.4 Thuật toán Logistic Regression**

Logistic Regression là một trong những thuật toán phân loại nhị phân cơ bản và phổ biến nhất trong học máy. Mặc dù tên gọi là hồi quy, nhưng bản chất của nó là phân loại – mô hình tính xác suất đầu ra thuộc về lớp 1, sau đó áp dụng ngưỡng để phân lớp.

Khác với Linear Regression cho kết quả là giá trị liên tục, Logistic Regression sử dụng hàm sigmoid để giới hạn đầu ra trong khoảng từ 0 đến 1, đại diện cho xác suất.



Hình 2. 7. Thuật toán Logistic Regression

Theo Géron, 2019, Logistic Regression sử dụng hàm sigmoid để chuyển đầu ra thành xác suất, phù hợp cho bài toán phân loại nhị phân [18].

$$S(z) = \frac{1}{1 + e^{-z}}$$

Dễ triển khai: Logistic Regression rất nhẹ, phù hợp với các hệ thống web đơn giản.

Giải thích được: Có thể biết biến nào ảnh hưởng nhiều đến kết quả → hữu ích trong y tế.

Không đòi hỏi quá nhiều dữ liệu lớn: Phù hợp với các tập dữ liệu vừa như Bangladeshi Dataset (~5000 mẫu).

Theo nghiên cứu của Faruque et al. (2021), Logistic Regression đạt độ chính xác lên đến 84.7% khi áp dụng vào bộ dữ liệu tiểu đường Bangladesh – một con số rất khả quan và phù hợp triển khai cộng đồng [15].

### 2.3.5 Các chỉ số đánh giá mô hình

Khi xây dựng mô hình học máy, không chỉ cần huấn luyện tốt, mà còn phải đánh giá mô hình bằng các chỉ số cụ thể:

**Accuracy (Độ chính xác):** Tỷ lệ dự đoán đúng trên toàn bộ dữ liệu.

**Precision (Độ chính xác dương):** Tỷ lệ người được mô hình dự đoán là có nguy cơ mắc bệnh và thật sự đúng là mắc bệnh.

**Recall (Tỷ lệ phát hiện đúng):** Tỷ lệ người mắc bệnh được mô hình nhận diện đúng.

**F1-Score:** Trung bình hài hòa giữa Precision và Recall.

**AUC – ROC Curve:** Đánh giá khả năng phân biệt của mô hình ở nhiều ngưỡng khác nhau, dùng cho phân tích sâu.

## 2.4 Tiền xử lý dữ liệu trong học máy

Trước khi áp dụng bất kỳ thuật toán học máy nào, dữ liệu cần được xử lý và chuẩn bị một cách kỹ lưỡng. Đây là một trong những bước quan trọng nhất quyết định đến độ chính xác và hiệu quả của mô hình. Trong nhiều trường hợp, chất lượng của dữ liệu còn quan trọng hơn bản thân thuật toán được sử dụng. Chính vì vậy, phần này sẽ trình bày một số bước tiền xử lý dữ liệu phổ biến trong học máy, đặc biệt áp dụng với các bài toán phân loại trong lĩnh vực y tế.

### 2.4.1 Làm sạch dữ liệu (Data Cleaning)

Dữ liệu thực tế, đặc biệt là dữ liệu y tế, thường có nhiều giá trị không hợp lệ, lỗi nhập liệu, hoặc thiếu thông tin. Làm sạch dữ liệu là quá trình loại bỏ hoặc xử lý các giá trị sai lệch để tránh ảnh hưởng xấu đến mô hình.

Ví dụ: Loại bỏ dòng có dữ liệu âm cho cân nặng, hoặc tuổi lớn hơn 120.

### 2.4.2 Xử lý giá trị thiếu (Missing Value Handling)

Trong nhiều tập dữ liệu, sẽ có các cột hoặc hàng bị thiếu giá trị. Một số cách phổ biến để xử lý là:

Loại bỏ dòng hoặc cột nếu số lượng giá trị thiếu quá nhiều.

Điền trung bình (mean) hoặc trung vị (median) cho dữ liệu số.



Điền giá trị phổ biến nhất (mode) cho dữ liệu phân loại.

Tùy theo tầm quan trọng của cột và tỉ lệ giá trị bị thiếu, người thực hiện sẽ đưa ra quyết định phù hợp.

Theo Han et al. (2011), xử lý giá trị thiếu là bước đầu tiên giúp đảm bảo dữ liệu không bị lệch phân phối hoặc gây sai số cho mô hình huấn luyện [19].

#### **2.4.3 Chuẩn hóa dữ liệu (Normalization / Standardization)**

Đối với dữ liệu số có đơn vị khác nhau (như đường huyết, huyết áp, BMI...), mô hình học máy có thể bị thiên lệch nếu không chuẩn hóa.

Normalization: biến đổi giá trị về khoảng (0,1).

Standardization: biến đổi dữ liệu về phân phối chuẩn với trung bình 0 và độ lệch chuẩn 1.

Việc chuẩn hóa giúp các biến có đóng góp công bằng trong mô hình, đặc biệt với Logistic Regression.

#### **2.4.4 Mã hóa nhãn (Label Encoding / One-hot Encoding)**

Nếu dữ liệu có biến dạng phân loại (categorical) như giới tính, nhóm tuổi, tiền sử... thì cần biến đổi thành dạng số để mô hình hiểu được.

Label Encoding: biến giá trị thành số nguyên (nam = 0, nữ = 1).

One-hot Encoding: tạo cột riêng cho từng giá trị (tránh tạo thứ bậc giả).

Việc chọn cách mã hóa nào phụ thuộc vào loại mô hình và đặc điểm dữ liệu.

Các kỹ thuật mã hóa nhãn phổ biến như label encoding và one-hot encoding thường được khuyến nghị trong bài toán phân loại [18].

#### **2.4.5 Phân chia tập huấn luyện và kiểm tra**

Dữ liệu sau khi tiền xử lý cần được chia làm hai tập riêng biệt:

**Tập huấn luyện (training set):** dùng để mô hình học từ dữ liệu.

**Tập kiểm tra (test set):** dùng để đánh giá khả năng tổng quát hóa.

## 2.5 Ngôn ngữ lập trình Python

### 2.5.1 Giới thiệu



*Hình 2. 8. Ngôn ngữ lập trình Python*

Python là một ngôn ngữ lập trình bậc cao, thông dịch, đa mục đích, được phát triển bởi Guido van Rossum và công bố lần đầu vào năm 1991. Với triết lý thiết kế “Code phải dễ đọc, dễ hiểu hơn là tối ưu hóa phức tạp”, Python nhanh chóng trở thành một trong những ngôn ngữ được ưa chuộng nhất trong giới lập trình, đặc biệt trong các lĩnh vực khoa học dữ liệu, trí tuệ nhân tạo, và phát triển ứng dụng web.

Theo khảo sát của Stack Overflow Developer Survey 2023, Python là một trong 3 ngôn ngữ phổ biến nhất được các nhà phát triển trên toàn thế giới sử dụng trong lĩnh vực khoa học dữ liệu và học máy [37].

Đồng thời, theo Python Software Foundation, Python được sử dụng bởi các công ty hàng đầu như Google, Netflix, Dropbox và NASA, nhờ khả năng đơn giản hóa các quy trình phức tạp bằng cú pháp gọn gàng [32].

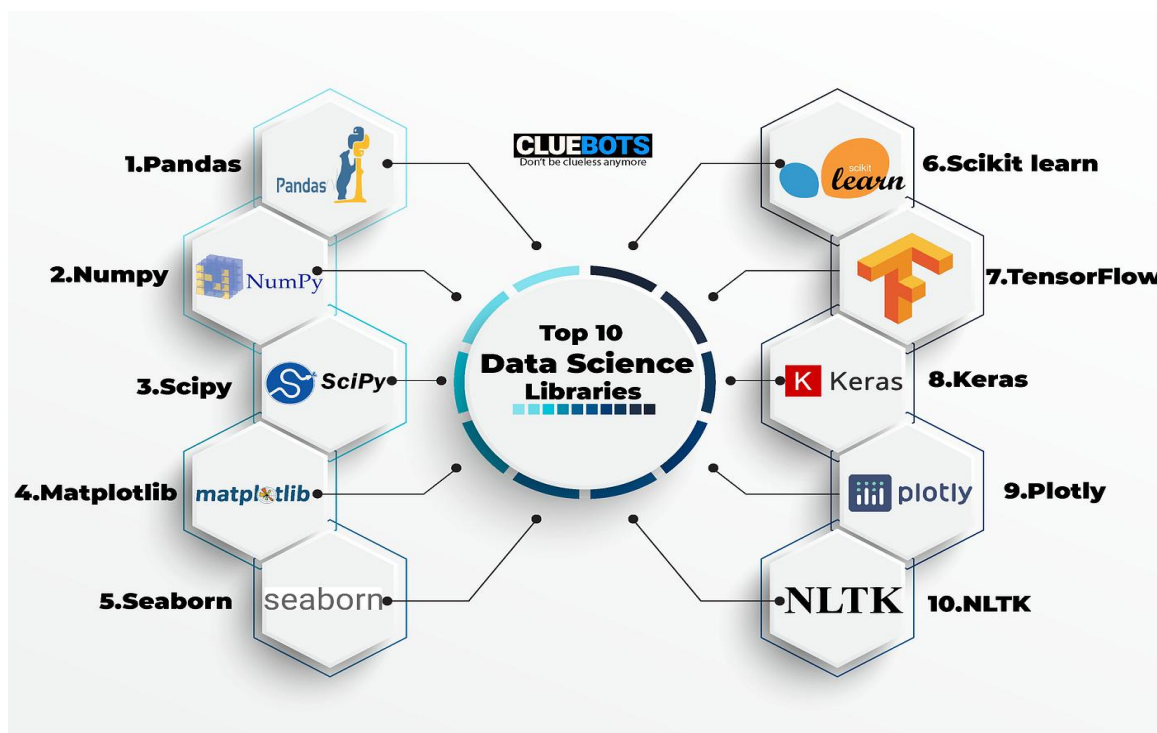
#### **Vai trò của Python trong đề tài:**

Trong khuôn khổ đề tài này, Python được lựa chọn làm ngôn ngữ chính để xử lý dữ liệu, huấn luyện mô hình học máy, và xây dựng API backend. Vì Python có thư viện học máy mạnh mẽ như scikit-learn, dễ dùng và phù hợp với Logistic Regression – thuật toán chính được sử dụng. Để kết nối với frontend thông qua Flask – một framework web nhẹ nhưng hiệu quả. Có cộng đồng hỗ trợ lớn và tài liệu hướng dẫn đầy đủ, dễ học ngay cả với sinh viên không chuyên sâu về lập trình.

## 2.5.2 Các thư viện của Python

Một trong những thế mạnh vượt trội của Python so với các ngôn ngữ khác là hệ sinh thái thư viện rất phong phú, đặc biệt trong các lĩnh vực như học máy, xử lý dữ liệu và trực quan hóa. Nhờ vào sự hỗ trợ mạnh mẽ của cộng đồng lập trình viên toàn cầu, Python có hàng loạt thư viện mã nguồn mở giúp người phát triển triển khai các thuật toán học máy một cách nhanh chóng, hiệu quả mà không cần phải xây dựng mọi thứ từ đầu.

Các thư viện Python phổ biến và phù hợp nhất để phục vụ cho từng giai đoạn: từ tiền xử lý dữ liệu, xây dựng mô hình học máy, đến trực quan hóa kết quả và kết nối với giao diện web. Các thư viện chính bao gồm:



Hình 2. 9. Các thư viện của Python

**Pandas:** Là thư viện mạnh mẽ hỗ trợ thao tác với dữ liệu dạng bảng (dataframe), tương tự như bảng Excel hoặc SQL. Hỗ trợ đọc/ghi dữ liệu từ nhiều định dạng (CSV, Excel, SQL...), xử lý thiếu dữ liệu, thống kê mô tả, lọc và nhóm dữ liệu. Là công cụ không thể thiếu trong tiền xử lý dữ liệu y tế vì dữ liệu đầu vào thường ở dạng bảng có hàng nghìn dòng và nhiều cột [31].

Numpy: Là thư viện tính toán mảng và đại số tuyến tính, dùng để xử lý các phép toán nền tảng cho học máy. Nhiều thư viện học máy như scikit-learn và TensorFlow đều được xây dựng dựa trên numpy [28].

Scikit-learn: Là thư viện học máy mã nguồn mở nổi tiếng, được phát triển từ năm 2007. Cung cấp đầy đủ các thuật toán phổ biến như: Logistic Regression, KNN, Decision Tree, Random Forest, SVM... Ngoài ra còn hỗ trợ chia tập dữ liệu (train/test split), chuẩn hóa dữ liệu, tính các chỉ số đánh giá như Accuracy, Precision, Recall, F1-score, AUC. Trong đề tài này, scikit-learn được dùng để triển khai mô hình Logistic Regression – một thuật toán phân loại nhị phân phù hợp với bài toán dự đoán nguy cơ mắc bệnh tiểu đường [34].

Matplotlib & Seaborn: Đây là hai thư viện dùng để trực quan hóa dữ liệu và kết quả dự đoán. Matplotlib hỗ trợ vẽ biểu đồ cơ bản như đường, cột, phân tán. Seaborn hỗ trợ vẽ các biểu đồ nâng cao đẹp mắt, dễ hiểu (ví dụ: heatmap, pairplot, boxplot...). Việc minh họa trực quan giúp người dùng không chuyên cũng hiểu được kết quả dự đoán và mối quan hệ giữa các biến [25].

Flask: là một framework web nhẹ cho Python, thường dùng để xây dựng các API. Trong đề tài, Flask được dùng để xây dựng API kết nối giữa frontend ReactJS và backend Python. Khi người dùng nhập dữ liệu, React sẽ gửi yêu cầu đến Flask API, và mô hình học máy trong Python sẽ trả về kết quả dự đoán [16].

### **2.5.3 Ưu điểm của Python trong triển khai học máy**

Việc lựa chọn Python làm ngôn ngữ chính trong đề tài không chỉ dựa trên sự phổ biến, mà còn xuất phát từ những ưu điểm kỹ thuật nổi bật mà Python mang lại cho quá trình phát triển hệ thống học máy nói chung và dự đoán bệnh tiểu đường nói riêng. Dưới đây là những lý do cụ thể khiến Python trở thành lựa chọn phù hợp và hiệu quả:

Cú pháp đơn giản, dễ học, dễ viết: Python có cú pháp gần với ngôn ngữ tự nhiên (pseudocode), giúp người lập trình dễ đọc, dễ hiểu và viết nhanh hơn so với các ngôn ngữ như Java hay C++. Điều này đặc biệt có lợi đối với sinh viên, người mới tiếp cận lập trình, hoặc khi làm việc trong thời gian ngắn như đồ án tốt nghiệp.

Theo khảo sát của Stack Overflow (2023), Python là ngôn ngữ được đánh giá là “dễ tiếp cận nhất” cho người học mới [37].

Hệ sinh thái thư viện mạnh mẽ và liên kết chặt chẽ: Python có các thư viện hỗ trợ đầy đủ cho tiền xử lý dữ liệu (pandas, numpy), học máy (scikit-learn, xgboost), trực quan hóa (matplotlib, seaborn), kết nối web/API (Flask, FastAPI), triển khai mô hình (joblib, pickle, Flask-RESTful).

Hỗ trợ cộng đồng lớn và nhiều tài liệu hướng dẫn: Python có một cộng đồng sử dụng toàn cầu cực kỳ mạnh mẽ, với hàng triệu người dùng đóng góp trên StackOverflow, Github, Medium, và các khóa học mở. Nhờ đó, mọi lỗi thường gặp đều dễ tra cứu, sinh viên có thể học từ tài liệu chính thức hoặc các khóa học miễn phí từ Coursera, Kaggle, Google Colab.

Theo PSF (Python Software Foundation), Python là ngôn ngữ được hỗ trợ nhiều nhất trong lĩnh vực AI hiện nay [32].

Phù hợp với học máy và trí tuệ nhân tạo: Python sinh ra để phục vụ cho khoa học tính toán. Với các đặc tính như: dễ thao tác với dữ liệu, hỗ trợ biểu đồ, trực quan hóa, tích hợp tốt với các thư viện C/C++ để tăng tốc.

#### **2.5.4 Nhược điểm của Python trong triển khai học máy cho đề tài**

Mặc dù Python là một ngôn ngữ linh hoạt, mạnh mẽ và rất phổ biến trong lĩnh vực học máy, tuy nhiên, giống như bất kỳ công nghệ nào khác, nó cũng tồn tại những hạn chế nhất định. Việc nhận diện rõ các nhược điểm này là điều cần thiết để đánh giá khách quan và có phương án khắc phục khi triển khai thực tế.

Tốc độ xử lý không cao: Python là ngôn ngữ thông dịch (interpreted language), tức là mỗi dòng lệnh được thực thi theo thời gian thực chứ không biên dịch trước như C/C++ hay Java. Điều này khiến tốc độ xử lý của Python chậm hơn đáng kể khi làm việc với các tác vụ cần tính toán phức tạp hoặc xử lý dữ liệu lớn.

Trong các ứng dụng học máy quy mô lớn, như xử lý ảnh y khoa, chuỗi gen hoặc dữ liệu triệu bản ghi, Python có thể trở nên kém hiệu quả nếu không kết hợp với các thư viện tối ưu hóa bằng C/C++.

Không tối ưu cho ứng dụng di động hoặc nhúng: Python không được thiết kế để phát triển ứng dụng di động gốc (native mobile) hoặc hệ thống nhúng. Điều

này hạn chế khả năng tích hợp mô hình học máy vào các thiết bị y tế cầm tay, app Android/iOS hoặc phần cứng chuyên dụng.

Khó kiểm soát bộ nhớ nếu lập trình thiếu kinh nghiệm: Do quản lý bộ nhớ tự động (garbage collection), Python có thể tiêu tốn nhiều RAM hơn so với các ngôn ngữ như C hoặc Go, đặc biệt nếu thao tác trên tập dữ liệu lớn mà không tối ưu mã nguồn.

### 2.5.5 Ứng dụng của Python

Python đóng vai trò trung tâm trong toàn bộ quy trình từ xử lý dữ liệu đến huấn luyện mô hình và triển khai dự đoán. Với sự hỗ trợ mạnh mẽ từ các thư viện học máy, trực quan hóa, và framework web, Python được sử dụng một cách linh hoạt, hiệu quả và dễ kiểm soát.

Tiền xử lý dữ liệu y tế: Sử dụng pandas để đọc dữ liệu từ file CSV, kiểm tra và xử lý các giá trị bị thiếu, loại bỏ dữ liệu nhiễu. Dùng numpy để thực hiện các phép biến đổi số học, chuẩn hóa dữ liệu (min-max scaling) và chuyển đổi kiểu dữ liệu phù hợp cho huấn luyện mô hình.

Xây dựng và huấn luyện mô hình học máy: Sử dụng scikit-learn để triển khai thuật toán Logistic Regression, là mô hình phù hợp cho bài toán phân loại nhị phân (có/không nguy cơ tiểu đường). Áp dụng kỹ thuật chia tập huấn luyện và kiểm tra (train\_test\_split) để đánh giá mô hình một cách khách quan.

Tính toán các chỉ số như Accuracy, Precision, Recall, F1-score để kiểm tra hiệu suất mô hình. Lưu mô hình huấn luyện thành file .pkl bằng joblib để sử dụng lại trong dự đoán thực tế.

Dự đoán nguy cơ tiểu đường: Mô hình Logistic Regression sau khi huấn luyện được dùng để dự đoán xác suất mắc bệnh tiểu đường cho từng trường hợp bệnh nhân mới. Python xử lý đầu vào từ người dùng, áp dụng mô hình, sau đó trả kết quả dưới dạng xác suất và phân loại nguy cơ (thấp / trung bình / cao).

Trực quan hóa kết quả: Kết quả dự đoán được biểu diễn bằng biểu đồ cột, biểu đồ tròn, hoặc thanh đánh giá rủi ro sử dụng thư viện matplotlib. Biểu đồ giúp người dùng không chuyên dễ dàng hiểu được mức độ nguy cơ của mình thông qua hình ảnh.

Xây dựng API giao tiếp với giao diện website : Dùng Flask để xây dựng các API RESTful. Khi người dùng nhập dữ liệu từ frontend (ReactJS), Flask sẽ tiếp nhận, xử lý, đưa vào mô hình học máy và trả về kết quả dự đoán. Flask chạy như một server backend đơn giản, nhẹ, phù hợp cho hệ thống ứng dụng demo hoặc triển khai nội bộ.

## 2.6 ReactJS

Ngoài việc xây dựng mô hình học máy bằng Python ở phía backend, giao diện người dùng cũng đóng vai trò không kém phần quan trọng trong việc tạo ra trải nghiệm thân thiện, trực quan và dễ sử dụng. Vì vậy, thư viện ReactJS được chọn làm nền tảng phát triển frontend (giao diện website) nhờ sự linh hoạt, hiện đại và hiệu suất cao.



*Hình 2. 10. ReactJS – công cụ phát triển giao diện người dùng*

ReactJS là một thư viện JavaScript mã nguồn mở do Facebook phát triển, lần đầu ra mắt vào năm 2013. Mục tiêu chính của React là tạo ra các giao diện người dùng tương tác một cách hiệu quả và nhanh chóng thông qua cơ chế DOM ảo (Virtual DOM).

### 2.6.1 Lý do chọn ReactJS

Trong các thư viện và framework frontend hiện nay (Vue, Angular, Svelte...), ReactJS nổi bật nhờ:

Cộng đồng sử dụng rộng rãi, tài liệu phong phú.

Dễ học, dễ mở rộng, phù hợp với cả người mới lẫn chuyên gia.

Kiến trúc thành phần (component-based) giúp dễ tái sử dụng code.

Dễ kết nối với backend API qua axios hoặc fetch.

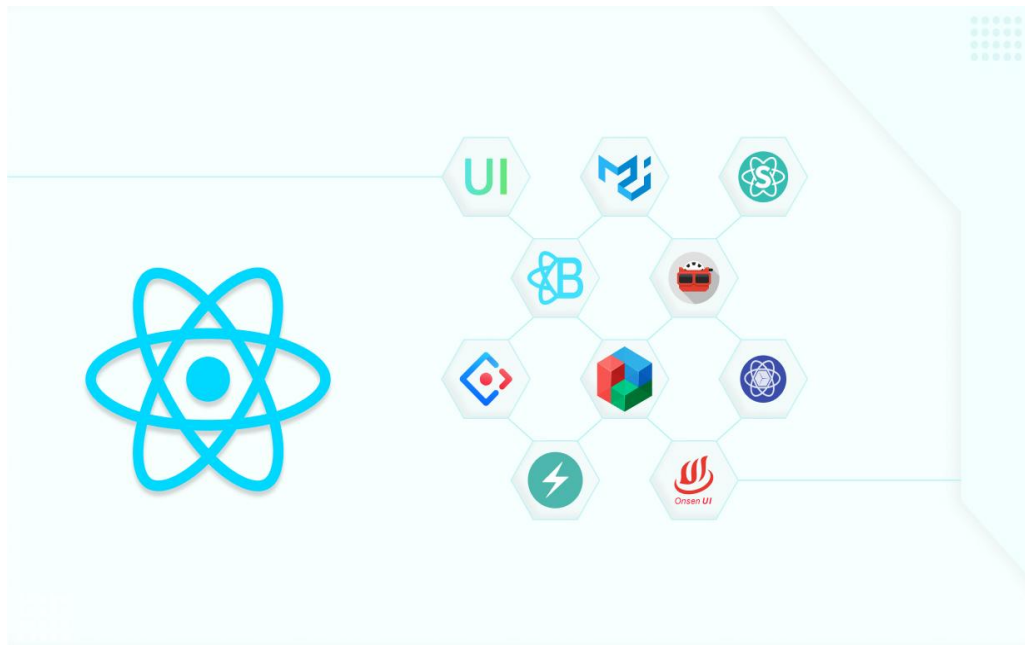
Do đó, ReactJS được chọn để triển khai giao diện trong đề tài nhằm đảm bảo:

Tốc độ phản hồi tốt,

Dễ duy trì, nâng cấp,

Có thể mở rộng lên PWA (Progressive Web App) trong tương lai nếu cần.

### 2.6.2 Các thư viện hỗ trợ sử dụng trong ReactJS



Hình 2. 11. Các thư viện trong ReactJS

Trong quá trình phát triển giao diện người dùng, đề tài còn tích hợp một số thư viện hỗ trợ phổ biến để tăng tốc độ xây dựng và cải thiện trải nghiệm người dùng:

react-router-dom: Theo tài liệu chính thức của React Router (2023), dùng để định tuyến (routing) giữa các trang trong ứng dụng như: Trang chủ, Trang dự đoán, Lịch sử dự đoán, Thông tin tài khoản. Giúp React chuyển trang mà không cần tải lại toàn bộ website (Single Page Application – SPA) [33].



axios: là thư viện phổ biến được dùng để gửi dữ liệu từ form về backend. Thư viện hỗ trợ gửi yêu cầu HTTP (GET, POST) từ frontend đến backend Flask API. Dùng để gửi dữ liệu người dùng nhập từ form đến server, nhận kết quả dự đoán trả về [9].

chart.js hoặc react-chartjs-2: Theo tài liệu của Chart.js (2023), dùng để hiển thị kết quả dự đoán dưới dạng biểu đồ (ví dụ: biểu đồ tròn, cột). Giao diện trực quan, dễ hiểu cho người dùng cuối, đặc biệt với kết quả phân loại theo cấp độ nguy cơ (thấp – trung bình – cao) [10].

tailwindcss: Theo tài liệu chính thức của Tailwind CSS (2023), dùng để xây dựng giao diện theo chuẩn responsive, tối ưu hóa hiển thị trên điện thoại và máy tính. Giúp tăng tốc độ thiết kế giao diện, thay vì phải viết CSS thủ công [38].

### **2.6.3 Ưu điểm của ReactJS**

Hiệu suất cao: Nhờ Virtual DOM, chỉ những phần bị thay đổi mới được cập nhật lại, website phản hồi nhanh.

Tái sử dụng được code: Các thành phần như form nhập liệu, biểu đồ, header/footer được tách thành các component, dễ tái sử dụng.

Dễ kết nối với backend: Gửi/nhận dữ liệu từ Flask API đơn giản qua axios.

Thân thiện người dùng: Giao diện rõ ràng, biểu đồ trực quan.

### **2.6.4 Nhược điểm của ReactJS**

Cần cấu hình ban đầu: Cài đặt môi trường React, cấu hình webpack, babel có thể gây khó khăn nếu chưa quen.

Quản lý trạng thái phức tạp nếu mở rộng: Khi hệ thống lớn dần, có thể cần đến Redux hoặc Context API để quản lý dữ liệu toàn cục.

Yêu cầu kiến thức JavaScript hiện đại: Cần hiểu ES6+, JSX, Hook...

### **2.6.5 Ứng dụng của ReactJS**

ReactJS đóng vai trò trung tâm trong việc xây dựng giao diện người dùng của hệ thống, giúp kết nối giữa người dùng và mô hình học máy một cách trực quan, thân thiện và hiệu quả. Trong đề tài này, React không chỉ đơn thuần là công

cụ hiển thị mà còn đóng vai trò trong tổ chức logic giao diện, xử lý luồng dữ liệu, hiển thị biểu đồ dự đoán, và quản lý trạng thái người dùng.

Dưới đây là các ứng dụng cụ thể của ReactJS trong hệ thống:

Xây dựng giao diện nhập liệu bệnh nhân: Tạo một form nhập thông tin bao gồm: tuổi, giới tính, chỉ số glucose, BMI, huyết áp, nhịp tim, tiền sử gia đình... Các trường nhập được kiểm tra bằng logic JavaScript (validation) ngay trên trình duyệt giúp tránh lỗi sai cơ bản. Dùng React state để lưu giá trị người dùng nhập, và chỉ khi hợp lệ mới cho gửi sang server.

Kết nối mô hình học máy qua API: Khi người dùng nhấn nút "Dự đoán", form sẽ gửi dữ liệu qua API Flask bằng axios. Sau khi server trả về xác suất nguy cơ mắc bệnh, giao diện sẽ hiển thị kết quả theo màu sắc cảnh báo (Xanh – Vàng – Đỏ) và biểu đồ cột minh họa trực quan.

Trực quan hóa kết quả bằng biểu đồ: Sử dụng thư viện chartjs để hiển thị xác suất dự đoán và biểu đồ so sánh các chỉ số y tế đã nhập với ngưỡng bình thường.

Ngoài ra còn các ứng dụng cụ thể của ReactJS trong thực tế hiện nay:

Mạng xã hội và nền tảng truyền thông: React được chính Meta (Facebook) phát triển và áp dụng đầu tiên vào giao diện người dùng của Facebook. Hiện nay, React là công nghệ cốt lõi phía frontend của Facebook, Instagram, WhatsApp Web.

Thương mại điện tử (E-commerce): React được nhiều trang thương mại điện tử lớn lựa chọn để tăng hiệu suất trải nghiệm người dùng như: Shopify, Walmart, Amazon clone, Nike.

Nền tảng học trực tuyến (E-learning): Các hệ thống giáo dục trực tuyến cần giao diện tương tác liên tục như: Coursera, Udemy, edX.

Dashboard quản lý nội bộ (CRM, ERP): Các doanh nghiệp vừa và lớn thường xây dựng hệ thống quản trị dữ liệu nội bộ như Airbnb, Atlassian, Netflix.

Ngân hàng, tài chính, ví điện tử như Paypal, Coinbase, Wise.

## **2.7 Hệ quản trị cơ sở dữ liệu MySQL**

Trong lĩnh vực công nghệ thông tin, dữ liệu luôn đóng vai trò trung tâm trong việc vận hành và duy trì hoạt động của các hệ thống phần mềm. Để lưu trữ, quản lý và truy xuất dữ liệu một cách hiệu quả, các hệ quản trị cơ sở dữ liệu (Database Management Systems – DBMS) đã được phát triển.

Trong đó, MySQL là một hệ quản trị cơ sở dữ liệu quan hệ (Relational DBMS – RDBMS) phổ biến, được ứng dụng rộng rãi trong cả học thuật lẫn công nghiệp phần mềm.

### **2.7.1 Khái niệm về hệ quản trị cơ sở dữ liệu**

Hệ quản trị cơ sở dữ liệu là phần mềm trung gian cho phép người dùng tương tác với dữ liệu một cách có tổ chức, an toàn và có kiểm soát. Thông qua DBMS, người dùng có thể thực hiện các thao tác như thêm mới dữ liệu (insert), truy vấn (query), cập nhật (update) và xóa (delete) – gọi chung là các thao tác CRUD (Create, Read, Update, Delete) [13].

Các hệ quản trị CSDL phổ biến hiện nay gồm:

- Quan hệ (Relational): như MySQL, PostgreSQL, Oracle, Microsoft SQL Server.
- Phi quan hệ (NoSQL): như MongoDB, Firebase, Cassandra.
- Dạng kết hợp (Multi-model DBMS): như ArangoDB, OrientDB.

Trong đó, hệ quản trị cơ sở dữ liệu quan hệ vẫn giữ vị trí chủ đạo trong nhiều lĩnh vực như thương mại điện tử, tài chính, quản lý y tế, giáo dục... nhờ khả năng tổ chức dữ liệu chặt chẽ theo mô hình bảng [11].

### 2.7.2 Giới thiệu tổng quan về MySQL



*Hình 2. 12. MySQL - Hệ quản trị cơ sở dữ liệu quan hệ*

MySQL là một hệ quản trị cơ sở dữ liệu quan hệ mã nguồn mở được phát triển bởi công ty MySQL AB từ năm 1995 và hiện nay thuộc sở hữu của Oracle Corporation. Đây là một trong những RDBMS phổ biến nhất trên thế giới, được sử dụng trong hàng triệu website và hệ thống thông tin.

MySQL hoạt động theo mô hình client-server: người dùng hoặc ứng dụng sẽ gửi các truy vấn đến server chứa dữ liệu, và kết quả trả về sẽ được xử lý và hiển thị ở phía người dùng.

MySQL hỗ trợ đầy đủ các thao tác SQL chuẩn như: SELECT, INSERT, UPDATE, DELETE, JOIN, GROUP BY, ORDER BY... dùng để xử lý dữ liệu hiệu quả trong hệ thống thông tin [26].

### 2.7.3 Các đặc điểm nổi bật của MySQL

Mã nguồn mở, miễn phí: Phiên bản MySQL Community Edition hoàn toàn miễn phí, rất phù hợp với sinh viên, startup và doanh nghiệp nhỏ. Người dùng có thể cài đặt, tùy chỉnh và sử dụng mà không cần trả phí bản quyền [26].

Tốc độ xử lý nhanh: MySQL được tối ưu để xử lý nhanh các truy vấn trên hệ thống dữ liệu vừa và nhỏ. Nó có thể phục vụ hàng trăm ngàn truy vấn mỗi giây trong môi trường thực tế [29].

Tương thích cao với nhiều hệ điều hành: MySQL hoạt động tốt trên Windows, Linux, macOS và có thể dễ dàng triển khai trên máy chủ ảo, container (Docker), hoặc các nền tảng cloud như AWS, Google Cloud [30].

Cộng đồng lớn và dễ học: Do phổ biến và dễ sử dụng, MySQL có hệ sinh thái tài liệu và cộng đồng hỗ trợ rất rộng. Nhiều công cụ quản trị như phpMyAdmin, MySQL Workbench giúp thao tác trực quan, phù hợp với cả người mới [32].

#### **2.7.4 Kiến trúc và mô hình dữ liệu trong MySQL**

MySQL tuân theo mô hình dữ liệu quan hệ, trong đó thông tin được tổ chức thành các bảng (tables). Mỗi bảng bao gồm các cột (fields) và hàng (records). Các bảng có thể liên kết với nhau thông qua:

- Khóa chính (Primary Key): xác định duy nhất từng bản ghi trong bảng
- Khóa ngoại (Foreign Key): liên kết bảng hiện tại với khóa chính của bảng khác

Ngoài ra, hệ quản trị MySQL còn hỗ trợ các ràng buộc toàn vẹn (Constraints) như: NOT NULL, UNIQUE, CHECK, DEFAULT để đảm bảo tính đúng đắn cho dữ liệu [13].

Các kiểu dữ liệu phổ biến được hỗ trợ trong MySQL bao gồm:

- Dữ liệu số: INT, FLOAT, DOUBLE
- Dữ liệu chuỗi: VARCHAR, TEXT
- Thời gian: DATE, DATETIME
- Logic: BOOLEAN [26]

#### **2.7.5 Ưu điểm và nhược điểm của MySQL**

Ưu điểm:

- Miễn phí, mã nguồn mở (bản Community Edition)
- Dễ học, dễ triển khai cho người mới bắt đầu

- Cộng đồng người dùng lớn, tài liệu phong phú
- Hỗ trợ chuẩn SQL và các thao tác ACID
- Có các công cụ trực quan như phpMyAdmin, MySQL Workbench hỗ trợ quản trị hiệu quả

Nhược điểm:

- Tính năng nâng cao còn hạn chế: MySQL thiếu một số tính năng mạnh của PostgreSQL như hỗ trợ dữ liệu không cấu trúc, hoặc khả năng mở rộng theo chiều ngang [12].
- Không hỗ trợ đầy đủ các thao tác JOIN nâng cao: Chẳng hạn, FULL OUTER JOIN không có sẵn trong bản Community Edition [30].

## **CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU**

### **3.1 Tổng quan về hệ thống**

#### **3.1.1 Mô tả hệ thống**

Hệ thống được xây dựng nhằm mục tiêu hỗ trợ người dùng có thể tự đánh giá nguy cơ mắc bệnh tiểu đường type 2 thông qua việc nhập các thông số y tế cá nhân cơ bản. Giao diện của hệ thống được thiết kế tối giản, trực quan và dễ thao tác, phù hợp với cả những người không có kiến thức chuyên sâu về công nghệ hoặc y học.

Khi sử dụng hệ thống, người dùng sẽ được yêu cầu nhập các thông tin đầu vào bao gồm:

- Tuổi (Age)
- Giới tính (Gender)
- Chỉ số đường huyết (Glucose)
- Huyết áp tâm thu (Systolic BP) và tâm trương (Diastolic BP)
- Chỉ số BMI (Body Mass Index)
- Tiền sử gia đình (Family history of diabetes)
- Nhịp tim (Pulse rate)
- Cao huyết áp (Hypertensive)
- Tiền sử cao huyết áp (Family hypertensive)
- Bệnh tim mạch (Cardiovascular disease)
- Đột quỵ (Stroke)

Sau khi nhập thông tin, hệ thống sẽ xử lý dữ liệu, đưa vào mô hình học máy đã được huấn luyện và trả về kết quả dự đoán: có nguy cơ (thấp – trung bình – cao) mắc bệnh tiểu đường, kèm theo xác suất dự đoán cụ thể là bao nhiêu phần trăm.

Đồng thời, hệ thống còn lưu trữ lịch sử dự đoán của từng người dùng để tiện cho việc theo dõi, tra cứu sau này. Tất cả các thao tác này được thực hiện hoàn toàn

tự động và phản hồi kết quả chỉ trong vòng vài giây, giúp tiết kiệm thời gian và hỗ trợ ra quyết định kịp thời.

### 3.1.2 Kiến trúc hệ thống

Hệ thống ứng dụng được xây dựng dựa trên kiến trúc client-server, kết hợp với công nghệ trí tuệ nhân tạo (AI) trong xử lý dự đoán. Mỗi thành phần đảm nhận một vai trò riêng biệt, giúp tổ chức hệ thống rõ ràng, dễ bảo trì và nâng cấp.

Frontend – ReactJS: Giao diện nhập dữ liệu và hiển thị kết quả.

Backend – Flask API: Tiếp nhận yêu cầu từ giao diện, xử lý logic và gọi mô hình học máy.

Mô hình học máy – Logistic Regression: Được huấn luyện từ dữ liệu thực tế để dự đoán nguy cơ.

Cơ sở dữ liệu – MySQL: Lưu trữ thông tin người dùng và lịch sử dự đoán.

### 3.1.3 Thành phần hệ thống

ReactJS – Giao diện người dùng (Frontend): là nơi người dùng tương tác trực tiếp với hệ thống. Giao diện hiển thị các thành phần như:

- Form nhập thông tin y tế
- Nút “Dự đoán”
- Biểu đồ hiển thị kết quả
- Trang đăng nhập, đăng ký, trang chủ
- Lịch sử dự đoán
- Toàn bộ giao diện được xây dựng theo kiến trúc Component-based, giúp dễ bảo trì và mở rộng.

Flask API – Server xử lý trung gian (Backend): Đóng vai trò là cầu nối giữa frontend, mô hình học máy và cơ sở dữ liệu. Nhận dữ liệu JSON từ frontend qua phương thức POST và tiền xử lý dữ liệu trước khi đưa vào mô hình. Sau khi nhận kết quả dự đoán, Flask thực hiện:

- Gửi lại dữ liệu cho frontend



- Ghi kết quả vào MySQL
- Flask cũng xử lý xác thực người dùng bằng mã OTP Email.

Mô hình học máy – Logistic Regression: Là mô hình phân loại nhị phân được huấn luyện từ trước. Được lưu dưới định dạng .pkl và nạp mỗi khi có dữ liệu cần dự đoán. Mô hình được huấn luyện bằng thư viện scikit-learn.

MySQL – Cơ sở dữ liệu lưu trữ: Lưu thông tin người dùng: tài khoản, email, thời gian tạo. Lưu lịch sử các lần dự đoán, gồm:

- Thông tin đầu vào
- Kết quả dự đoán

Thiết kế CSDL tuân theo mô hình quan hệ, có khóa chính – khóa ngoại giữa bảng users và predictions.

### **3.1.4 Quy trình hoạt động**

Bước 1: Người dùng nhập dữ liệu đầu vào từ giao diện ReactJS.

Bước 2: Dữ liệu được gửi từ ReactJS sang Flask qua API RESTful.

Bước 3: Flask sẽ tiến xử lý dữ liệu và truyền dữ liệu vào mô hình học máy.

Bước 4: Mô hình Logistic Regression trả về kết quả dự đoán.

Bước 5: Flask gửi kết quả về lại ReactJS để hiển thị cho người dùng.

Bước 6: Flask lưu kết quả và dữ liệu vào cơ sở dữ liệu.

Bước 7: Người dùng có thể xem lịch sử các lần dự đoán trước đó.

## **3.2 Mô tả chức năng hệ thống**

### **3.2.1 Giao diện người dùng**

Thiết kế giao diện dành cho mọi đối tượng từ sinh viên, nhân viên văn phòng đến những người trung niên. Bố cục rõ ràng, phân chia từng phần nhập dữ liệu, nút dự đoán, hiển thị kết quả, biểu đồ.

Màu sắc trực quan dùng xanh – vàng - đỏ để thể hiện rõ mức độ nguy cơ mắc bệnh. Sử dụng ngôn ngữ tiếng Việt dễ hiểu, không sử dụng thuật ngữ kỹ thuật

phức tạp. Thân thiện với thiết bị di động: thiết kế responsive giúp hiển thị tốt trên cả máy tính và điện thoại.

### **3.2.2 Quản lý tài khoản**

Cho phép người dùng đăng nhập bằng email, mật khẩu.

Đăng ký tài khoản xác thực bằng mã OTP gửi đến Gmail.

Quên mật khẩu xác thực lại để đổi thành mật khẩu mới.

Kiểm tra quyền truy cập vào hệ thống, chỉ người dùng đã đăng nhập mới được sử dụng chức năng dự đoán và xem lịch sử.

### **3.2.3 Nhập thông tin y tế**

Người dùng có thể nhập các chỉ số sức khỏe cơ bản để tiến hành dự đoán, bao gồm: tuổi, chỉ số đường huyết, huyết áp tâm thu và tâm trương, chỉ số BMI, tiền sử gia đình. Đây là chức năng trung tâm để đưa ra kết quả dự đoán.

### **3.2.4 Dự đoán và hiển thị kết quả**

Sau khi nhập các thông tin y tế theo yêu cầu và bấm nút “Dự đoán” sẽ trả về kết quả bao nhiêu phần trăm mắc bệnh kèm theo biểu đồ. Kết quả dự đoán sẽ được hiển thị câu thông báo và biểu đồ dạng cột để so sánh chỉ số của người dùng với mức bình thường.

### **3.2.5 Lưu và xem lại lịch sử dự đoán**

Mỗi lần dự đoán sẽ được lưu lại tự động vào cơ sở dữ liệu bao gồm:

Dữ liệu mà người dùng đã nhập vào, kết quả dự đoán, thời gian thực hiện, tài khoản người dùng tương ứng và biểu đồ. Chức năng này giúp người dùng có thể theo dõi sự thay đổi sức khỏe của mình theo thời gian.

Người dùng có thể xem lại các lần dự đoán trước đó. Danh sách theo dạng bảng bao gồm ngày giờ, kết quả, dữ liệu đầu vào. Có thể xem lại chi tiết biểu đồ và xóa nếu cảm thấy quá nhiều lịch sử.

### 3.3 Quy trình xử lý dữ liệu y tế

#### 3.3.1 Giới thiệu tập dữ liệu sử dụng

Trong đề tài này, em sử dụng bộ dữ liệu có tên “Bangladeshi Dataset for Type 2 Diabetes”, được công bố trên nền tảng Kaggle và là một trong những bộ dữ liệu y tế thực tế có độ tin cậy cao cho nghiên cứu dự đoán tiểu đường type 2.

**Số dòng (bệnh nhân):** 5437

**Số cột (thuộc tính):** 15

**Định dạng:** CSV

**Nguồn gốc:** Dữ liệu được thu thập từ các bệnh viện tại Bangladesh, đã qua xử lý sơ bộ.[9]

#### 3.3.2 Các thuộc tính chính trong tập dữ liệu

*Bảng 3. 1. Mô tả các thuộc tính trong tập dữ liệu*

Thuộc tính	Mô tả
Age	Tuổi bệnh nhân
Gender	Giới tính (Nam/Nữ)
Pulse rate	Nhịp tim
Sytolic BP	Huyết áp tâm thu
Diastolic BP	Huyết áp tâm trương
Glucose	Chỉ số đường huyết
Height	Chiều cao
Weight	Cân nặng
BMI	Chỉ số khối cơ thể
Family Diabetes	Tiền sử bệnh tiểu đường trong gia đình

Hypertensive	Cao huyết áp
Family Hypertensive	Tiền sử cao huyết áp trong gia đình
Cardiovascular Disease	Bệnh tim mạch
Stroke	Đột quỵ
Diabetic	Nhãn đầu ra Xem người đó có bị tiểu đường hay không

### 3.3.1 Các bước xử lý dữ liệu

Dữ liệu y tế ban đầu được xử lý và chuẩn bị thông qua hai giai đoạn chính: làm sạch và chuẩn hóa. Các bước cụ thể được thực hiện như sau:

#### **Bước 1: Đọc dữ liệu và ép kiểu chính xác**

Tập dữ liệu **diabetes\_final\_data\_v2.csv** được đọc bằng thư viện pandas.

Các cột số nguyên như age, pulse\_rate, systolic\_bp, diastolic\_bp được ép kiểu về dạng **Int64**.

Các cột số thực như glucose, bmi, height, weight được ép kiểu về float để đảm bảo tính chính xác khi xử lý và huấn luyện mô hình.

#### **Bước 2: Mã hóa biến phân loại**

Cột gender được mã hóa: Male = 1, Female = 0

Cột diabetic là nhãn mục tiêu được chuyển từ Yes/No thành 1/0

#### **Bước 3: Phân chia tập huấn luyện và kiểm tra**

Dữ liệu được chia thành hai phần theo tỷ lệ 80% train / 20% test

Dùng tham số stratify=y để đảm bảo tỷ lệ nhãn (0/1) được phân bổ đều trong cả hai tập.

#### **Bước 4: Cân bằng dữ liệu bằng SMOTE**

Dữ liệu gốc bị mất cân bằng (số người không mắc bệnh nhiều hơn rất nhiều), do đó kỹ thuật SMOTE (Synthetic Minority Over-sampling Technique) được áp dụng để:

Tạo thêm dữ liệu giả lập cho lớp thiểu số

Áp dụng chỉ trên tập huấn luyện, không làm thay đổi tập kiểm tra

#### **Bước 5: Chuẩn hóa dữ liệu bằng StandardScaler**

Tất cả đặc trưng được chuẩn hóa về phân phối chuẩn với trung bình = 0, độ lệch chuẩn = 1

Giúp mô hình Logistic Regression học tốt hơn và hội tụ nhanh hơn.

### **3.4 Xây dựng và huấn luyện học máy**

#### **3.4.1 Lý do chọn Logistic Regression**

Trong bài toán phân loại nhị phân (0 – không mắc bệnh, 1 – mắc bệnh tiểu đường), Logistic Regression là thuật toán phù hợp vì:

Hoạt động tốt với dữ liệu tuyến tính và không yêu cầu tập dữ liệu quá lớn.

Dễ huấn luyện, diễn giải và giải thích với người dùng không chuyên.

Có thể xuất xác suất dự đoán – giúp người dùng hiểu rõ mức độ rủi ro.

Là nền tảng trong các hệ thống y tế vì tính minh bạch cao.

#### **3.4.2 Mô hình Logistic Regression trong scikit-learn**

Mô hình được khởi tạo và huấn luyện bằng thư viện scikit-learn, với các tham số:

`random_state=42`: đảm bảo tính tái lập

`max_iter=1000`: cho phép mô hình hội tụ trong trường hợp dữ liệu phức tạp

#### **3.4.3 Lưu mô hình bằng joblib**

Sau khi huấn luyện, mô hình sẽ được lưu lại thành file .pkl bằng thư viện joblib để tái sử dụng khi dự đoán không cần huấn luyện lại và giúp backend (Flask) gọi nhanh chóng khi người dùng gửi dữ liệu.

### 3.4.4 Đánh giá mô hình

Sau khi huấn luyện, mô hình được đánh giá trên tập kiểm tra ( $X_{test\_scaled}$ ,  $y_{test}$ ) bằng các chỉ số:

Độ chính xác (Accuracy)

Ma trận nhầm lẫn (Confusion Matrix)

Báo cáo phân loại (Classification Report): Precision (độ chính xác cao theo từng lớp), Recall (khả năng mô hình phát hiện đúng người mắc bệnh), F1-score (trung bình điều hòa giữa Precision và Recall).

### 3.4.5 Nhận xét về hiệu quả mô hình

Mô hình Logistic Regression cho kết quả tốt, độ chính xác cao. Sau khi xử lý dữ liệu (SMOTE với chuẩn hóa), mô hình hoạt động ổn định và ít overfitting. Kết quả dự đoán có thể diễn giải được cho người dùng vì logistic regression đơn giản và trực quan.

## 3.5 Thiết kế cơ sở dữ liệu

### 3.5.1 Lý do chọn MySQL

Là hệ quản trị cơ sở dữ liệu quan hệ mã nguồn mở phổ biến, dễ sử dụng. Hỗ trợ tốt thao tác CRUD (Create – Read – Update – Delete). Tương thích với PHP, Python (Flask), và các công cụ như XAMPP, phpMyAdmin. Cộng đồng lớn, tài liệu phong phú, phù hợp với sinh viên và hệ thống quy mô vừa.

### 3.5.2 Mô tả các bảng dữ liệu

*Bảng 3. 2. Mô tả các bảng dữ liệu*

Số thứ tự	Tên thực thể	Mô tả
1.	users	Tài khoản người dùng
2.	predictions	Lưu lịch sử dự đoán

### 3.5.3 Mô tả thuộc tính của các bảng

*Bảng 3. 3. Mô tả thuộc tính bảng users*

STT	Tên tắt thuộc tính	Diễn giải	Loại giá trị	Kiểu dữ liệu	Miền giá trị	Chiều dài	Ghi chú
1.	id	id	Bắt buộc	Int	Khóa chính	11	
2.	email	email	Bắt buộc	Varchar		100	
3.	password	Mật khẩu	Bắt buộc	Vachar		255	
4.	username	Tên đăng nhập	Bắt buộc	Vachar		100	
5.	reset code	Mã khôi phục mật khẩu		Vachar		10	
6.	reset code expiry	Thời gian hết hạn mã		datetime			
7.	created at	Ngày tạo		Timestamp			

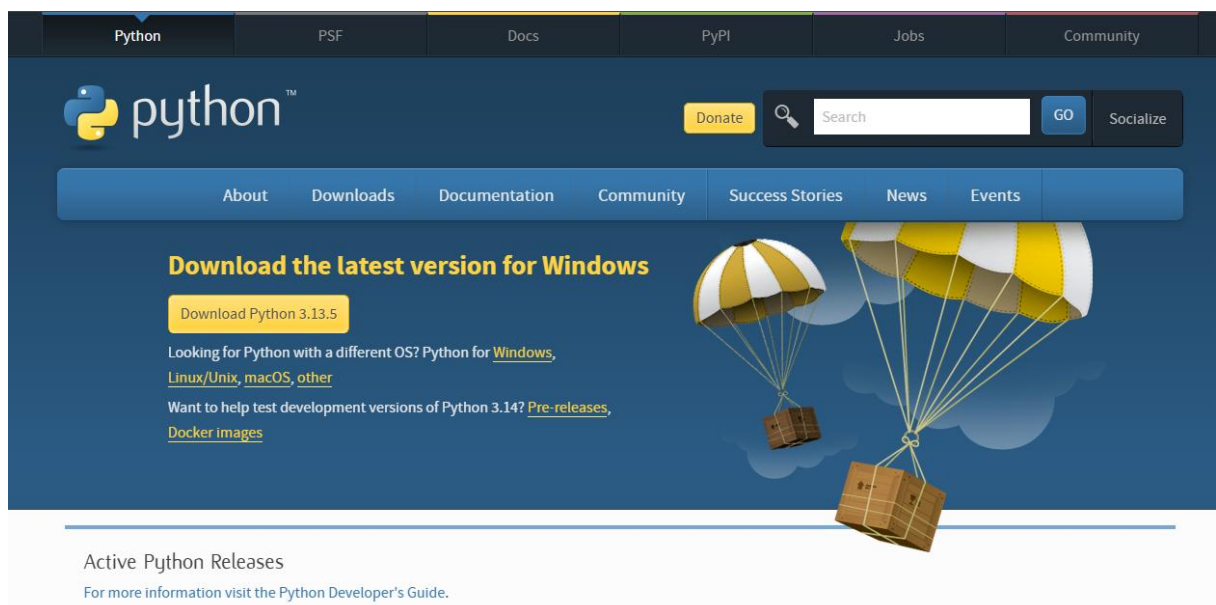
*Bảng 3. 4. Bảng mô tả thuộc tính bảng predictions*

STT	Tên tắt thuộc tính	Diễn giải	Loại giá trị	Kiểu dữ liệu	Miền giá trị	Chiều dài	Ghi chú
1.	id	id	Bắt buộc	Int	Khóa chính	11	
2.	user_id	Id người dùng	Bắt buộc	Int	Khóa ngoại	11	
3.	age	Tuổi	Bắt buộc	Int		11	
4.	gender	Giới tính	Bắt buộc	Tinyint		4	
5.	pulse rate	Nhịp tim	Bắt buộc	Int		11	
6.	sytolic bp	Huyết áp tâm thu	Bắt buộc	Int		11	
7.	diastolic bp	Huyết áp tâm trương	Bắt buộc	Int		11	
8.	glucose	Chỉ số đường huyết	Bắt buộc	Float			
9.	height	Chiều cao	Bắt buộc	Float			

STT	Tên tắt thuộc tính	Diễn giải	Loại giá trị	Kiểu dữ liệu	Miền giá trị	Chiều dài	Ghi chú
10.	weight	Cân nặng	Bắt buộc	Float			
11.	bmi	Chỉ số khối cơ thể	Bắt buộc	Float			
12.	family diabetes	Tiền sử gia đình	Bắt buộc	Tinyint		4	
13.	hypertensive	Tăng huyết áp	Bắt buộc	Tinyint		4	
14.	family hypertensive	Tiền sử tăng huyết áp	Bắt buộc	Tinyint		4	
15.	cardiovascular disease	Bệnh tim mạch	Bắt buộc	Tinyint		4	
16.	stroke	Đột quy	Bắt buộc	Tinyint		4	
17.	prediction result	Kết quả dự đoán	Bắt buộc	Varchar		100	
18.	prediction probability	Xác suất dự đoán	Bắt buộc	Float			
19.	created at	Ngày tạo	Bắt buộc	datetime			

### 3.6 Cài đặt và chạy môi trường Python

#### 3.6.1 Cài đặt Python



Hình 3. 1. Giao diện nơi tải về để cài đặt Python



Tải Python từ trang chính thức: <https://www.python.org>

Cài đặt phiên bản Python 3.10 hoặc mới hơn

Trong quá trình cài, đánh dấu “Add Python to PATH”

### 3.6.2 Cài đặt các thư viện cần thiết

**pip install flask**

**pip install flask-cors**

**pip install pandas**

**pip install numpy**

**pip install scikit-learn**

**pip install joblib**

**pip install imbalanced-learn**

Các thư viện này phục vụ cho: Flask API, huấn luyện và chuẩn hóa mô hình, cân bằng dữ liệu với SMOTE.

### 3.6.3 Chạy Flask API

Tạo file app.py (nơi sẽ chứa xử lý yêu cầu)

Mở file app.py

Mở terminal và chạy lệnh **python app.py**

Flask sẽ khởi chạy tại địa chỉ: <http://localhost:5000>

## 3.7 Cài đặt và triển khai giao diện ReactJS

### 3.7.1 Tạo project ReactJS

Mở Visual Studio Code lên, nhấn mở Terminal và gõ dòng lệnh như sau:

**npx create-react-app frontend**

Lệnh này sẽ tạo một thư mục tên **frontend**.

Trong lúc cài đặt, thư mục frontend sẽ chứa đầy đủ dự án, cấu trúc các thư mục con, các file bên trong.



Sau khi cài đặt xong, di chuyển vào thư mục frontend.

Gõ lệnh: **cd frontend**

Từ đó có thể bắt đầu xây dựng giao diện với ReactJS.

### 3.7.2 Cài đặt các thư viện cần thiết

**npm install axios**

**npm install react-router-dom**

**npm install recharts**

**npm install react-toastify**

**npm install react-modal**

Trong đó:

- **axios:** gửi request đến Flask backend
- **react-router-dom:** điều hướng giữa các trang
- **recharts:** hiển thị biểu đồ kết quả
- **react-toastify:** hiển thị thông báo nhẹ nhàng

### 3.7.3 Chạy thử giao diện

**npm start**

Trang sẽ mở ở: <http://localhost:3000>. Tự động reload khi sửa code (hot reload). Nếu backend chạy ở cổng 5000, hai hệ thống frontend–backend có thể giao tiếp qua localhost.

## 3.8 Cài đặt XAMPP cho MySQL

### 3.8.1 Cài đặt XAMPP

Truy cập trang chủ: <https://www.apachefriends.org>

Tải phiên bản XAMPP phù hợp với hệ điều hành (Windows)

Cài đặt bình thường → Sau khi hoàn tất, mở XAMPP Control Panel

The screenshot shows the 'Download' section of the Apache Friends website. It features a navigation bar with links like 'Download', 'Hosting', 'Community', and 'About'. Below the navigation bar, the word 'Download' is prominently displayed. The main content area includes a description of XAMPP as an easy-to-install Apache distribution. A table lists three versions of XAMPP for Windows (8.0.30, 8.1.25, and 8.2.12) with their respective checksums and sizes. To the right of the table, there is a 'Documentation/FAQs' section with links to Linux, Windows, and OS X FAQs. At the bottom of the table, there are links for 'Requirements' and 'More Downloads'.

Version	Checksum	Size
8.0.30 / PHP 8.0.30	md5 sha1	144 Mb
8.1.25 / PHP 8.1.25	md5 sha1	148 Mb
8.2.12 / PHP 8.2.12	md5 sha1	149 Mb

Hình 3. 2. Giao diện nơi tải về để cài đặt XAMPP

Nhấn Start cho dịch vụ:

- Apache (nếu dùng PHP)
- MySQL (bắt buộc để dùng phpMyAdmin)

### 3.8.2 Truy cập phpMyAdmin

Trên trình duyệt, truy cập địa chỉ: <http://localhost/phpmyadmin>

Giao diện quản trị MySQL sẽ xuất hiện. Có thể tạo database mới và thao tác trực quan với các bảng.

### 3.8.3 Tạo cơ sở dữ liệu

Bước 1: Tại tab “Cơ sở dữ liệu” nhập tên CSDL, ví dụ: diabetes\_app

Bước 2: Nhấn nút **Tạo**

Bước 3: Tạo bảng users và predictions

### 3.8.4 Kiểm tra kết nối với Python

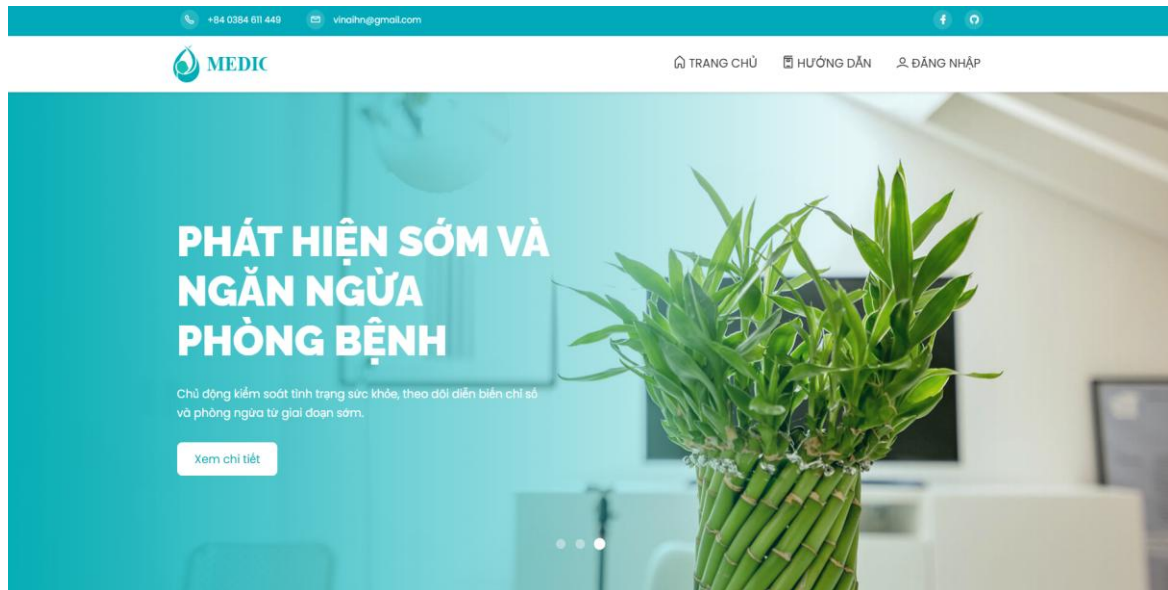
Cài đặt và sử dụng thư viện **mysql-connector-python** hoặc **pymysql** trong Flask để kết nối và thao tác dữ liệu, gõ lệnh:

**pip install mysql-connector-python**

## CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU

### 4.1 Giao diện trang chủ

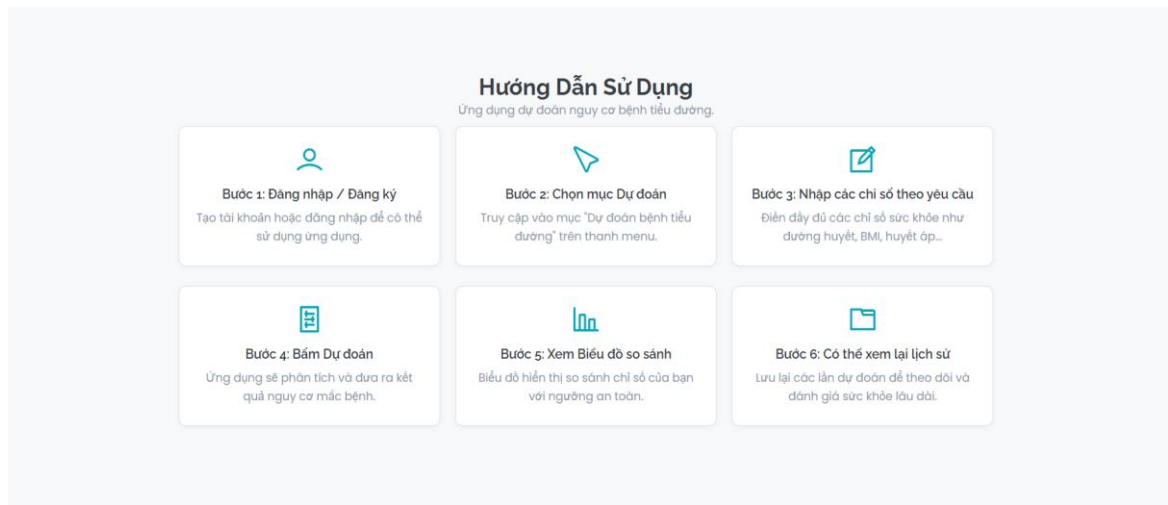
Giao diện trang chủ là điểm khởi đầu đầu tiên khi người dùng truy cập vào hệ thống. Giao diện được thiết kế hiện đại, đơn giản và thân thiện, nhằm tạo ấn tượng chuyên nghiệp và giúp người dùng dễ dàng định hướng các chức năng chính ngay từ đầu.



Hình 4. 1. Giao diện trang chủ

Thanh điều hướng (Navbar): Phần đầu giao diện là thanh điều hướng nằm ngang, nổi bật với logo hệ thống, đường dây nóng hỗ trợ và các nút chức năng bao gồm: Trang chủ, Hướng dẫn sử dụng, Đăng nhập/Đăng ký.

Phần tiêu đề chính (Banner đầu trang): Ngay bên dưới là phần tiêu đề truyền thông với khẩu hiệu “Phát hiện sớm và ngăn ngừa phòng bệnh”.



*Hình 4. 2. Giao diện Hướng dẫn sử dụng hệ thống*

Mục Hướng dẫn sử dụng hệ thống trình bày các bước sử dụng như sau:

Bước 1: Đăng nhập/ Đăng ký tài khoản

Bước 2: Chọn mục Dự đoán

Bước 3: Nhập các chỉ số y tế theo yêu cầu

Bước 4: Nhấn nút “Dự đoán”

Bước 5: Xem biểu đồ kết quả và xác suất

Bước 6: Lưu và có thể xem lại lịch sử dự đoán

Trải nghiệm người dùng: Trang chủ không chỉ giới thiệu chức năng mà còn đảm nhận vai trò định hướng giúp người dùng biết hệ thống sử dụng như thế nào, nắm rõ quy trình sử dụng và bắt đầu thao tác.

## 4.2 Giao diện đăng nhập và đăng ký

Hệ thống cung cấp chức năng đăng nhập và đăng ký nhằm đảm bảo tính cá nhân hóa và bảo mật dữ liệu người dùng. Mỗi người dùng sau khi đăng nhập sẽ có quyền truy cập vào các chức năng như: nhập thông tin y tế để dự đoán, xem lại lịch sử, xóa lịch sử, và chỉnh sửa tài khoản của chính mình.

The screenshot shows a login form titled "Đăng Nhập" (Login). It contains the following elements:

- A title "Đăng Nhập" in bold blue text.
- An email input field containing "vinaihn@gmail.com".
- A password input field with masked characters "\*\*\*\*\*" and a "SHOW" link to its right.
- A blue button labeled "Đăng Nhập" (Login).
- A blue link "Quên mật khẩu?" (Forgot password?).
- A text prompt "Chưa có tài khoản? Đăng ký" (Don't have an account? Register) with a blue link "Đăng ký" (Register).
- A link "← Quay về Trang chủ" (← Back to Home page).

Hình 4. 3. Giao diện đăng nhập

Giao diện đăng nhập: Người dùng nhập email và mật khẩu đã đăng ký trước đó. Nút “Đăng nhập” giúp xác thực thông tin và chuyển hướng vào hệ thống nếu hợp lệ. Ngoài ra, giao diện còn có liên kết trở lại trang chủ và chuyển hướng sang giao diện đăng ký.

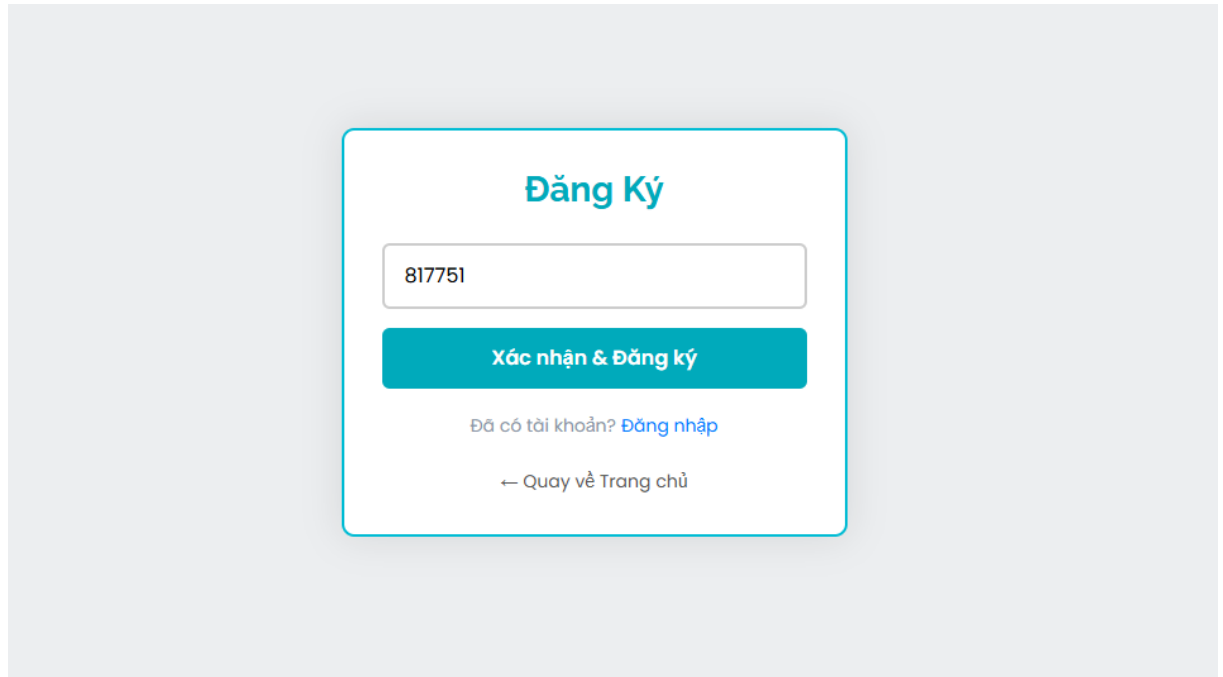
The screenshot shows a registration form titled "Đăng Ký" (Register). It contains the following elements:

- A title "Đăng Ký" in bold blue text.
- A name input field containing "Nguyễn Đăng".
- An email input field containing "nguyentinovn@gmail.com".
- A password input field with masked characters "\*\*\*\*\*" and a "SHOW" link to its right.
- A second password input field with masked characters "\*\*\*\*\*" and a "SHOW" link to its right.
- A blue button labeled "Gửi mã xác thực" (Send verification code).
- A text prompt "Đã có tài khoản? Đăng nhập" (Already have an account? Login) with a blue link "Đăng nhập" (Login).
- A link "← Quay về Trang chủ" (← Back to Home page).

Hình 4. 4. Giao diện đăng ký

Giao diện đăng ký: Người dùng mới có thể đăng ký bằng cách nhập họ tên, email cá nhân, mật khẩu. Sau khi điền đầy đủ thông tin, hệ thống sẽ gửi mã xác thực (OTP) về email để xác minh.

Chỉ khi mã xác thực đúng thì tài khoản mới được tạo thành công. Điều này giúp tăng mức độ an toàn cho hệ thống và tránh việc tạo sử dụng tài khoản ảo.

The image shows a web interface for registration confirmation. It features a white rectangular box with rounded corners and a thin blue border, centered on a light gray background. At the top of the box, the text "Đăng Ký" is displayed in a bold, teal font. Below this, there is a white input field containing the number "817751". Underneath the input field is a solid teal button with the white text "Xác nhận & Đăng ký". Below the button, the text "Đã có tài khoản? Đăng nhập" is shown in a smaller, teal font. At the bottom of the box, there is a link "← Quay về Trang chủ" in a small, gray font.

*Hình 4. 5. Giao diện xác thực mã OTP để đăng ký*

Chức năng Quên mật khẩu: Để hỗ trợ người dùng trong trường hợp không nhớ mật khẩu đăng nhập, hệ thống cung cấp chức năng "Đặt lại mật khẩu" thông qua mã xác thực OTP gửi về email.

Tại giao diện “Đặt Lại Mật Khẩu” người dùng chỉ cần nhập đúng địa chỉ email đã đăng ký. Sau đó, hệ thống sẽ tự động gửi mã xác thực về địa chỉ email đã nhập và hiển thị giao diện nhập mã.



**Đặt Lại Mật Khẩu**

vinaihn@gmail.com

**Gửi mã xác thực**

[← Quay lại đăng nhập](#)

*Hình 4. 6. Giao diện đặt lại mật khẩu*

Sau khi nhập mã xác thực thành công sẽ chuyển hướng đến giao diện đổi lại mật khẩu mới. Việc đặt lại mật khẩu có giới hạn thời gian hiệu lực cho mỗi mã xác thực, đảm bảo an toàn và tránh lạm dụng.

**Xác Thực Mã**

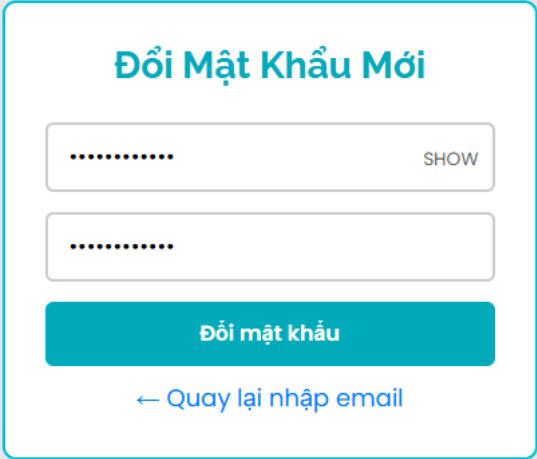
737583

**Xác thực**

[← Quay lại nhập email](#)

*Hình 4. 7. Giao diện xác thực mã OTP*

Nếu người dùng nhập sai mã hoặc để mã hết hạn, hệ thống sẽ thông báo lỗi và yêu cầu thực hiện lại quy trình. Chức năng này giúp hệ thống trở nên chuyên nghiệp, thân thiện hơn và đảm bảo người dùng không bị mất quyền truy cập vào tài khoản do quên mật khẩu.



Hình 4. 8. Giao diện nhập mật khẩu mới

### 4.3 Giao diện trang dự đoán

Giao diện trang dự đoán là chức năng trọng tâm của hệ thống, nơi người dùng nhập các thông số y tế cá nhân để hệ thống tiến hành phân tích và dự đoán nguy cơ mắc bệnh tiểu đường type 2.

Giao diện nhập dữ liệu y tế: Được thiết kế theo dạng biểu mẫu rõ ràng, trực quan, sử dụng ngôn ngữ Tiếng Việt dễ hiểu, giúp người dùng phổ thông cũng có thể thao tác mà không gặp khó khăn.

Các trường nhập bao gồm: tuổi, giới tính, nhịp tim, chỉ số đường huyết (mmol/L), chiều cao, cân nặng, chỉ số BMI, huyết áp tâm thu và tâm trương, tiền sử gia đình mắc bệnh tiểu đường, tiền sử tăng huyết áp, bệnh tim mạch, tiền sử đột quỵ. Tất cả các trường đều được thể hiện bằng select box hoặc input dạng số, giúp kiểm soát định dạng dữ liệu và hạn chế sai sót trong quá trình nhập liệu.

**DỰ ĐOÁN NGUY CƠ BỆNH TIỂU ĐƯỜNG**

Nhập các chỉ số theo yêu cầu để bắt đầu dự đoán

Tuổi Nhập tuổi	Giới tính -- Chọn giới tính --	Nhịp tim Nhập nhịp tim	Huyết áp tâm thu Nhập huyết áp tâm thu	Huyết áp tâm trương Nhập huyết áp tâm trương
Chỉ số đường huyết (mmol/L) Nhập chỉ số đường huyết	Chiều cao (m) Nhập chiều cao (m)	Cân nặng (kg) Nhập cân nặng (kg)	Chỉ số BMI (Cân nặng / Chiều cao <sup>2</sup> )	Tiền sử gia đình -- Chọn --
Tăng huyết áp -- Chọn --	Tiền sử huyết áp cao trong gia đình -- Chọn --	Bệnh tim mạch -- Chọn --	Tiền sử đột quỵ -- Chọn --	

**Dự đoán**

Hình 4. 9. Giao diện trang chức năng Dự đoán

#### 4.4 Giao diện kết quả dự đoán

Sau khi người dùng nhập đầy đủ thông tin y tế và nhấn nút “Dự đoán”, hệ thống sẽ gửi dữ liệu đến mô hình Logistic Regression thông qua Flask API. Mô hình xử lý và trả kết quả dự đoán theo thời gian thực.

**DỰ ĐOÁN NGUY CƠ BỆNH TIỂU ĐƯỜNG**

Nhập các chỉ số theo yêu cầu để bắt đầu dự đoán

Tuổi 50	Giới tính Nữ	Nhịp tim (nhịp/phút - bpm) 76	Huyết áp tâm thu (mmHg) 130	Huyết áp tâm trương (mmHg) 85
Chỉ số đường huyết (mmol/L) 7.2	Chiều cao (m) 1.70	Cân nặng (kg) 75	Chỉ số BMI (kg/m <sup>2</sup> ) 25.95	Tiền sử gia đình Không
Tăng huyết áp Không	Tiền sử huyết áp cao trong gia đình Không	Bệnh tim mạch Có	Tiền sử đột quỵ Không	

**Dự đoán**

**Kết quả dự đoán**

Đánh giá mức nguy cơ: ● CAO -> Có nguy cơ mắc bệnh tiểu đường - ● CAO (Xác suất: 76.8%).

**Xem biểu đồ so sánh**

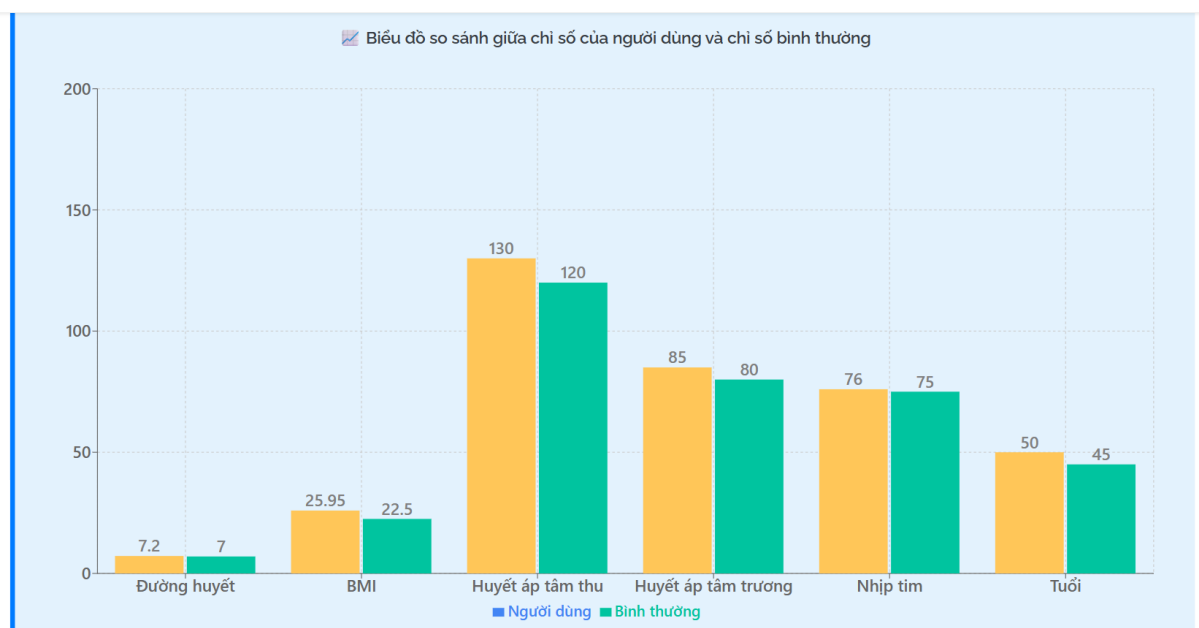
Hình 4. 10. Giao diện dự đoán trả kết quả

Hiện thị kết quả dự đoán dạng văn bản: Ngay bên dưới biểu mẫu nhập liệu, hệ thống sẽ hiển thị mức độ nguy cơ, xác suất cụ thể, màu sắc cảnh báo ( xanh – thấp, vàng – trung bình, đỏ - cao). Ngoài ra, hệ thống thể hiện rõ mức độ nguy cơ cho người dùng để chủ động kiểm tra sức khỏe tại các cơ sở y tế nếu kết quả rơi vào vùng cảnh báo.

Hiện thị biểu đồ so sánh chỉ số: Hệ thống cung cấp thêm biểu đồ dạng cột để trực quan hóa dữ liệu đã nhập. Mỗi cột biểu diễn một chỉ số y tế như: đường huyết, huyết áp, nhịp tim, BMI. Có cột so sánh với mức chỉ số bình thường của cơ thể.

Dữ liệu do người dùng nhập được tô màu nổi bật để dễ nhận biết phần chênh lệch. Biểu đồ giúp người dùng dễ dàng nhìn thấy chỉ số nào đang ở ngưỡng nguy hiểm hoặc vượt mức cho phép, từ đó có cơ sở điều chỉnh chế độ ăn uống và lối sống.

Giao diện này không chỉ cung cấp kết quả nhanh chóng, rõ ràng mà còn đóng vai trò giáo dục sức khỏe, giúp người dùng nâng cao nhận thức về các chỉ số y tế quan trọng.



Hình 4. 11. Giao diện xem kết quả dạng biểu đồ

#### 4.5 Giao diện lịch sử dự đoán

Giao diện lịch sử dự đoán cho phép người dùng đã đăng nhập có thể xem lại toàn bộ các lần họ đã thực hiện dự đoán bệnh tiểu đường trước đó.

Chức năng lịch sử dự đoán là một phần quan trọng trong hệ thống, giúp người dùng theo dõi, quản lý và đánh giá lại các lần dự đoán nguy cơ mắc bệnh tiểu đường của bản thân theo thời gian. Không chỉ phục vụ mục tiêu tra cứu, giao diện này còn là công cụ giúp người dùng chủ động nhận diện theo dõi sức khỏe.

Danh sách lịch sử dự đoán: Hệ thống hiển thị bảng dữ liệu với các cột thông tin bao gồm: thời gian (ngày/ giờ dự đoán), tuổi, giới tính, chỉ số glucose, chỉ số BMI, kết quả xác suất, chức năng xem lại chi tiết hiển thị biểu đồ từng lần dự đoán và có thể xóa từng bản ghi lịch sử.

Mỗi dòng dữ liệu tương ứng với một lần người dùng thực hiện dự đoán. Hệ thống sắp xếp các bản ghi theo thứ tự mới nhất lên trên để tiện theo dõi.

Chức năng thao tác trên bảng: Ngoài khả năng xem thông tin, người dùng còn có thể tương tác với từng bản ghi qua các nút chức năng. Có thể xem biểu đồ hiển thị trực quan dữ liệu của bản ghi đã chọn bằng biểu đồ cột.

Điều này giúp người dùng dễ dàng nhận ra chỉ số nào đang cao, thấp hay ổn định. Xem chi tiết hiển thị lại đầy đủ dữ liệu đầu vào và kết quả phân tích.

Xóa lịch sử nếu người dùng không muốn lưu giữ bản ghi, có thể xóa vĩnh viễn. Hệ thống hiển thị cảnh báo xác nhận trước khi thực hiện để tránh thao tác sai.

<div>  <div> <a href="#">TRANG CHỦ</a> <a href="#">HƯỚNG DẪN</a> <a href="#">DỰ ĐOÁN</a> <a href="#">HÀO NGUYỄN</a> </div> <div> <div>Chọn ngày</div> <div>Chọn tháng</div> <div>Chọn nam</div> </div> </div>								
Ngày	Tuổi	Giới tính	Đường huyết	BMI	Nhịp tim	Nguy cơ	Xác suất	Thao tác
00:17 17/06/2025	50	Nữ	7,2	25,95	76	Cao	76.8%	<a href="#">Chi tiết</a> <a href="#">Xóa</a>
22:20 16/06/2025	50	Nam	7,2	25,95	76	Trung bình	55.8%	<a href="#">Chi tiết</a> <a href="#">Xóa</a>
21:46 16/06/2025	50	Nữ	7,2	25,95	76	Trung bình	52.0%	<a href="#">Chi tiết</a> <a href="#">Xóa</a>
21:37 16/06/2025	50	Nữ	7,2	25,95	76	Cao	76.8%	<a href="#">Chi tiết</a> <a href="#">Xóa</a>
11:33 16/06/2025	50	Nam	7,2	25,95	76	Trung bình	55.8%	<a href="#">Chi tiết</a> <a href="#">Xóa</a>
22:56 15/06/2025	50	Nam	7,2	25,95	76	Trung bình	55.8%	<a href="#">Chi tiết</a> <a href="#">Xóa</a>
22:55 15/06/2025	30	Nam	5	20,76	70	Thấp	19.3%	<a href="#">Chi tiết</a> <a href="#">Xóa</a>
22:44 15/06/2025	50	Nam	7,2	25,95	76	Cao	75.3%	<a href="#">Chi tiết</a> <a href="#">Xóa</a>
22:44 15/06/2025	80	Nam	7,2	25,95	76	Cao	62.4%	<a href="#">Chi tiết</a> <a href="#">Xóa</a>
18:26 08/06/2025	65	Nam	9,5	34,89	80	Cao	99.7%	<a href="#">Chi tiết</a> <a href="#">Xóa</a>

Hình 4. 12. Giao diện Trang lịch sử dự đoán

Giao diện biểu đồ chi tiết từng lần dự đoán: Khi người dùng nhấn vào nút “Xem biểu đồ”, hệ thống sẽ hiển thị biểu đồ so sánh giữa chỉ số của người dùng với mức bình thường theo tiêu chuẩn y tế. Biểu đồ hỗ trợ nhận diện mức độ lệch chuẩn, qua đó cảnh báo tình trạng sức khỏe một cách dễ hiểu.

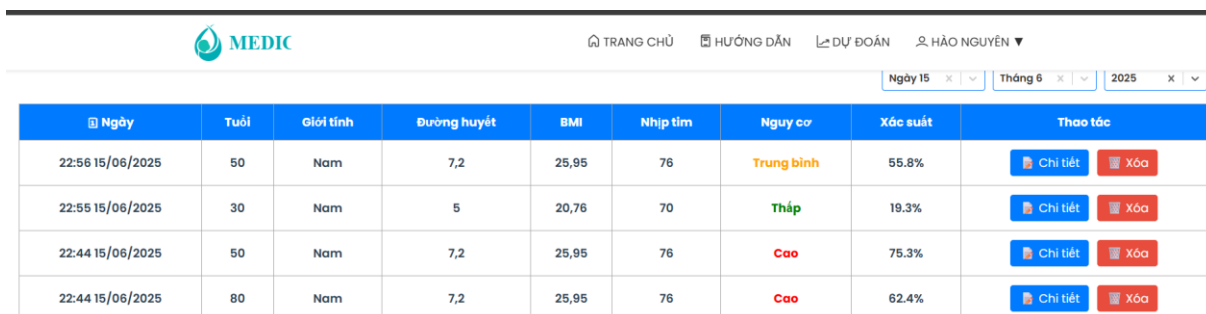


Hình 4. 13. Giao diện xem lại chi tiết lịch sử

Giá trị sử dụng thực tiễn: Chức năng lịch sử không chỉ phục vụ nhu cầu theo dõi sức khỏe cá nhân, mà còn giúp người dùng đối chiếu tất cả kết quả giữa nhiều thời điểm (ví dụ: trước và sau khi thay đổi chế độ ăn uống), khuyến khích người dùng duy trì thói quen theo dõi định kỳ, thay vì chờ có triệu chứng mới đi khám.

## 4.6 Chức năng lọc danh sách lịch sử theo thời gian

Để giúp người dùng dễ dàng tra cứu các lần dự đoán cũ trong khoảng thời gian mong muốn, hệ thống cung cấp tính năng lọc lịch sử dự đoán theo ngày – tháng – năm. Đây là một chức năng quan trọng khi số lượng bản ghi trong lịch sử ngày càng nhiều và người dùng cần xem lại các dữ liệu cụ thể.



The screenshot shows the MEDIC web application interface. At the top, there is a navigation bar with the MEDIC logo and links for 'TRANG CHỦ', 'HƯỚNG DẪN', 'DỰ ĐOÁN', and a user profile 'HÀO NGUYỄN'. Below the navigation bar, there are three dropdown filters for 'Ngày 15', 'Tháng 6', and '2025'. The main content is a table with the following columns: 'Ngày' (Date), 'Tuổi' (Age), 'Giới tính' (Gender), 'Đường huyết' (Blood Sugar), 'BMI', 'Nhịp tim' (Heart Rate), 'Nguy cơ' (Risk), 'Xác suất' (Probability), and 'Thao tác' (Actions). The table contains four rows of data, each with a date, age, gender, blood sugar, BMI, heart rate, risk level, and probability, along with 'Chi tiết' (Details) and 'Xóa' (Delete) buttons.

Ngày	Tuổi	Giới tính	Đường huyết	BMI	Nhịp tim	Nguy cơ	Xác suất	Thao tác
22:56 15/06/2025	50	Nam	7,2	25,95	76	Trung bình	55.8%	<a href="#">Chi tiết</a> <a href="#">Xóa</a>
22:55 15/06/2025	30	Nam	5	20,76	70	Thấp	19.3%	<a href="#">Chi tiết</a> <a href="#">Xóa</a>
22:44 15/06/2025	50	Nam	7,2	25,95	76	Cao	75.3%	<a href="#">Chi tiết</a> <a href="#">Xóa</a>
22:44 15/06/2025	80	Nam	7,2	25,95	76	Cao	62.4%	<a href="#">Chi tiết</a> <a href="#">Xóa</a>

Hình 4. 14. Lọc lịch sử theo ngày, tháng, năm

Giao diện lọc thời gian: Ngay phía trên bảng lịch sử, hệ thống hiển thị chọn ngày, tháng, năm. Người dùng có thể chọn chỉ ngày (xem lịch sử trong một ngày cụ thể), chỉ tháng và năm (xem toàn bộ dự đoán trong tháng và năm đó) hoặc kết hợp cả ba để lọc chính xác cho từng mốc thời gian mong muốn. Giao diện được thiết kế với dropdown tiện lợi, hỗ trợ chọn nhanh và hiển thị ngay kết quả lọc bên dưới.

Kết quả sau khi lọc: Ngay khi người dùng thay đổi bất kỳ giá trị nào trong ba bộ lọc, hệ thống sẽ tự động lọc và hiển thị ngay kết quả bên dưới, mà không cần tải lại trang. Những bản ghi không phù hợp sẽ tự động ẩn đi, chỉ giữ lại danh sách các bản ghi thuộc mốc thời gian đã chọn.

Tốc độ cập nhật nhanh, mượt mà giúp người dùng cảm thấy hệ thống phản hồi tốt, đặc biệt hữu ích khi lịch sử chứa nhiều dữ liệu. Nhờ vậy, người dùng có thể dễ dàng kiểm tra lại kết quả trong một đợt khám sức khỏe cụ thể, hoặc theo dõi sự

thay đổi trước và sau một giai đoạn điều trị, chẳng hạn như sau khi thay đổi chế độ ăn hoặc bắt đầu sử dụng thuốc.

Đánh giá tổng quan và giá trị thực tiễn: Chức năng lọc lịch sử dự đoán theo thời gian không chỉ là một tính năng kỹ thuật, mà còn đóng vai trò là công cụ quản lý sức khỏe cá nhân hiệu quả theo thời gian. Người dùng không cần ghi nhớ thủ công mà vẫn có thể xem được toàn bộ tiến trình chỉ số y tế của mình một cách có hệ thống.

Giao diện lọc được thiết kế rõ ràng, thân thiện với mọi đối tượng người dùng, kể cả người lớn tuổi. Tính năng này không chỉ hữu ích về mặt chức năng mà còn thể hiện sự quan tâm đến trải nghiệm thực tế của người sử dụng, một yếu tố cần thiết cho bất kỳ hệ thống chăm sóc sức khỏe nào muốn triển khai lâu dài.



## CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

### 5.1 Kết luận

Khóa luận “ "Xây dựng ứng dụng phân tích dữ liệu y tế để dự đoán nguy cơ mắc bệnh tiểu đường” trong quá trình thực hiện, vừa nghiên cứu lý thuyết về bệnh tiểu đường, các yếu tố ảnh hưởng, thuật toán mô hình Logistic Regression, vừa xây dựng ứng dụng, xử lý dữ liệu, huấn luyện mô hình đến giao diện website và lưu trữ dữ liệu dự đoán.

Xây dựng được giao diện người dùng bằng ReactJS bao gồm: các trang đăng nhập, đăng ký, dự đoán, lịch sử dự đoán, biểu đồ.

Thiết kế biểu đồ trực quan, dễ hiểu, phân bố màu sắc cụ thể theo mức độ nguy cơ mắc bệnh.

Xử lý logic, kết nối mô hình và trả kết quả theo thời gian thực.

Xây dựng cơ sở dữ liệu để lưu thông tin và các lần dự đoán.

.Huấn luyện mô hình học máy để dự đoán nguy cơ mắc bệnh dựa trên các chỉ số thông tin của người dùng.

Toàn bộ các chức năng chính đều đã hoạt động: Người dùng có thể nhập chỉ số y tế, nhận kết quả, xem biểu đồ minh họa, xem lại lịch sử và lọc theo thời gian. Tốc độ phản hồi nhanh, giao diện dễ dùng, dữ liệu được lưu đầy đủ.

Dù thời gian có hạn và vẫn còn một số điểm cần tối ưu thêm, nhưng em cảm thấy hài lòng vì đã tự mình làm được một hệ thống đầy đủ, rõ ràng từ frontend đến backend, có mô hình học máy và kết nối hoàn chỉnh.

### 5.2 Hướng phát triển

Mặc dù hệ thống đã hoàn thành và vận hành ổn định với các chức năng chính, nhưng trong quá trình triển khai thực tế và đánh giá thử nghiệm, em nhận thấy còn nhiều tiềm năng để phát triển, mở rộng cũng như tối ưu hiệu quả hệ thống trong tương lai.

Định hướng phát triển có thể chia thành 3 nhóm chính: nâng cấp kỹ thuật, mở rộng tính năng người dùng, và ứng dụng thực tế trong cộng đồng y tế.

### 5.2.1 Hướng phát triển về mặt kỹ thuật

Đa dạng hóa mô hình học máy: Hiện tại hệ thống chỉ sử dụng Logistic Regression, tuy phù hợp với dữ liệu tuyến tính nhưng còn hạn chế trong xử lý dữ liệu phi tuyến. Do đó, có thể mở rộng huấn luyện thêm các mô hình mạnh hơn như: Random Forest, XGBoost giúp tăng độ chính xác, xử lý tốt dữ liệu phức tạp, Neural Network (MLPClassifier) để mô phỏng cơ bản, phù hợp cho các mô hình tự học nâng cao.

Việc bổ sung nhiều mô hình cũng mở ra khả năng so sánh kết quả theo từng mô hình, cho phép người dùng tự lựa chọn thuật toán mà họ tin tưởng hoặc phù hợp hơn.

Nâng cấp giao diện người dùng (UI/UX): Dù giao diện hiện tại đơn giản và dễ sử dụng, nhưng trong tương lai có thể thiết kế thêm giao diện dạng thẻ y tế điện tử cho mỗi lần dự đoán, cho phép tải kết quả xuống dạng PDF, bổ sung giao diện bác sĩ để xem được kết quả của nhiều bệnh nhân. Tối ưu responsive cho các thiết bị di động có màn hình nhỏ hơn.

Tự huấn luyện mô hình định kỳ (Auto Retrain): Sau khi hệ thống được sử dụng thật, có thể tích hợp cơ chế thu thập dữ liệu thực tế (có gán nhãn). Từ đó, định kỳ sau một thời gian nhất định (1–2 tháng), hệ thống tự động tái huấn luyện mô hình để ngày càng tối ưu và cập nhật theo xu hướng mới.

Triển khai cloud, REST API công khai: Để mở rộng quy mô sử dụng, hệ thống có thể được triển khai lên nền tảng cloud như Heroku, Render, hoặc server riêng (VPS).

Ngoài ra, tạo REST API công khai sẽ giúp các hệ thống khác (ví dụ: phần mềm bệnh viện) có thể tích hợp và gửi dữ liệu trực tiếp đến mô hình để lấy kết quả dự đoán.

App mobile đa nền tảng: Sử dụng React Native hoặc Flutter để xây dựng ứng dụng điện thoại, giúp những người không quen dùng máy tính vẫn dễ dàng sử dụng trên smartphone. Ứng dụng này có thể có thêm chức năng gửi thông báo nhắc lịch kiểm tra định kỳ.

### **5.2.2 Hướng phát triển về chức năng người dùng**

Bên cạnh việc nâng cấp kỹ thuật, hệ thống cũng có thể mở rộng thêm nhiều tính năng phục vụ trực tiếp cho người dùng, nhằm tăng tính cá nhân hóa và khả năng theo dõi sức khỏe dài hạn.

Một định hướng quan trọng là xây dựng hệ thống phân quyền người dùng, bao gồm: người dùng cá nhân (người bệnh), bác sĩ (hoặc chuyên viên y tế), và quản trị viên hệ thống. Mỗi nhóm sẽ có giao diện và chức năng riêng biệt.

Người dùng thông thường sẽ nhập dữ liệu, nhận kết quả và theo dõi lịch sử cá nhân. Bác sĩ có thể xem nhiều bệnh nhân, đối chiếu chỉ số và đưa ra đánh giá y tế. Quản trị viên có quyền giám sát hệ thống, thống kê mô hình, xử lý tài khoản và cập nhật mô hình học máy khi cần thiết.

Ngoài ra, hệ thống có thể mở rộng tính năng phân tích lịch sử dự đoán, hiển thị theo dạng biểu đồ hoặc bảng thống kê. Người dùng có thể xem lại sự thay đổi các chỉ số như glucose, BMI, huyết áp theo tuần, tháng hoặc quý.

Hệ thống cũng có thể tự động phát hiện xu hướng xấu đi và đưa ra cảnh báo. Ví dụ, nếu chỉ số đường huyết liên tục tăng trong 3 lần gần nhất, hệ thống sẽ khuyến nghị người dùng đi khám sớm hoặc điều chỉnh chế độ sinh hoạt. Những gợi ý này giúp người dùng không chỉ biết mình có nguy cơ, mà còn hiểu cần phải làm gì tiếp theo.

### **5.2.3 Hướng phát triển theo hướng cộng đồng – y tế**

Về lâu dài, hệ thống có thể mở rộng và áp dụng trong môi trường thực tế như: trạm y tế, phòng khám, nhà thuốc, hoặc các chương trình sàng lọc sức khỏe cộng đồng.

Tích hợp hệ thống với thiết bị y tế thực tế, như: máy đo đường huyết, huyết áp, đồng hồ thông minh (smartwatch)... Việc kết nối sẽ giúp người dân lấy dữ liệu trực tiếp mà không cần nhập tay, từ đó tăng độ chính xác và tiết kiệm thời gian thao tác.

Ngoài ra, hệ thống cũng có thể phát triển thành kiosk tra cứu sức khỏe tự động:

Cài đặt hệ thống trên máy tính bảng hoặc màn hình cảm ứng, đặt ở nơi công cộng (siêu thị, nhà thuốc...). Người dân nhập số liệu y tế, nhận kết quả dự đoán và hướng dẫn trực tiếp tại chỗ.

Kiosk này có thể đặt ở những khu vực nông thôn, nơi người dân chưa quen với công nghệ hoặc không có điện thoại thông minh. Chỉ cần vài thao tác đơn giản như chọn tuổi, cân nặng, chiều cao và chỉ số đường huyết (nếu đo được), người dùng sẽ nhận kết quả trực tiếp, hiển thị bằng màu sắc và ngôn ngữ dễ hiểu.

Việc này giúp xóa bỏ rào cản công nghệ, mở rộng phạm vi tiếp cận đến nhiều đối tượng hơn – đặc biệt là người lớn tuổi hoặc những người có nguy cơ cao nhưng không có điều kiện tiếp cận y tế hiện đại.

Thậm chí, có thể kết hợp với nhân viên y tế cộng đồng để hỗ trợ nhập liệu cho người dân nếu họ không biết thao tác. Hệ thống cũng có thể in ra phiếu kết quả tóm tắt sau khi dự đoán, ghi rõ tình trạng nguy cơ, khuyến nghị nên làm gì tiếp theo. Phiếu này có thể được mang đến trạm y tế để bác sĩ tham khảo khi khám bệnh.

Về lâu dài, nếu triển khai ở quy mô lớn hơn, hệ thống còn có thể kết hợp cùng các chương trình khám sức khỏe định kỳ hoặc chiến dịch sàng lọc bệnh tiểu đường diện rộng tại địa phương. Trong những chiến dịch như vậy, thay vì ghi phiếu tay, nhân viên có thể dùng máy tính bảng tích hợp sẵn hệ thống này, nhập số liệu trực tiếp và ra kết quả nhanh chóng – vừa tiết kiệm thời gian, vừa tạo ra dữ liệu có cấu trúc để phân tích tổng thể sau này.

## PHỤ LỤC

## DANH MỤC TÀI LIỆU THAM KHẢO

### Tiếng Việt

1. Bộ Y tế. Hướng dẫn chẩn đoán và điều trị nội tiết – chuyển hóa (3879/QĐ-BYT). Hà Nội: NXB Y học, 2014.
2. Bộ Y tế. Hướng dẫn chẩn đoán và điều trị đái tháo đường típ 2 (Quyết định số 3319/QĐ-BYT). Hà Nội, 2017
3. Bộ Y tế. Báo cáo quốc gia về sức khỏe học sinh Việt Nam năm 2021. Hà Nội, 2021.
4. Bộ Y tế. Báo cáo quốc gia về chi phí điều trị bệnh tiểu đường tại Việt Nam, 2021.
5. Tổ chức Y tế Thế giới (WHO) (2021). *Sự thật chính về bệnh đái tháo đường*.
6. Tổ chức Y tế Thế giới (WHO) (2023). *Thống kê toàn cầu về tử vong và gánh nặng bệnh tật do đái tháo đường*.

### Tiếng Anh

7. American Diabetes Association (2023), *Statistics About Diabetes*, ADA, Arlington.
8. Apple Inc. (2022). *About Apple Health*
9. Axios (2023), *Axios HTTP Client for the Browser and Node.js*, Axios Project.
10. Chart.js (2023), *Chart.js Official Documentation*, Chart.js Developers.
11. Coronel, C., & Morris, S. (2019), *Database Systems: Design, Implementation, & Management* (13th ed.), Cengage Learning.
12. DigitalOcean (2022). *PostgreSQL vs MySQL: Key Differences*, DigitalOcean Docs.
13. Elmasri, R., & Navathe, S. B. (2015), *Fundamentals of Database Systems* (7th ed.), Pearson Education, Boston.
14. FDA (2021). *Digital Health Devices approved for diabetes care*
15. Faruque, M. O., Rahman, M. M., & Sultana, T. (2021), “Performance Analysis of Machine Learning Models in Diabetes Prediction using

- Bangladesh Dataset”, *International Journal of Computer Applications*, 183(28), pp. 1–6.
16. Flask Documentation (2023), *Flask Web Framework Official Docs*, Pallets Projects.
  17. Glucose Buddy. (2021). *Glucose Buddy mobile app – Overview*
  18. Géron, A. (2019), *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow* (2nd ed.), O'Reilly Media, Sebastopol, CA.
  19. Han, J., Kamber, M., Pei, J. (2011), *Data Mining: Concepts and Techniques*, Morgan Kaufmann.
  20. Harvard Medical School. *Diabetes Risk and Family History*. Harvard Health Publishing, 2021.
  21. Harvard Medical School. *Pulse Rate and Diabetes Risk*, Harvard Health, 2020.
  22. International Diabetes Federation (2021), *IDF Diabetes Atlas – 10th edition*, Brussels.
  23. James, G., Witten, D., Hastie, T., Tibshirani, R. (2013), *An Introduction to Statistical Learning*, Springer.
  24. Matplotlib (2023), *Matplotlib Documentation*, The Matplotlib Development Team.
  25. Mayo Clinic. *Sleep and Diabetes Risk*, 2020.
  26. MySQL (2023), *MySQL 8.0 Reference Manual*, Oracle Corporation.
  27. MySugr GmbH. (2022). *MySugr Diabetes App – Official Overview*
  28. NumPy Developers (2024), *NumPy Documentation*.
  29. Oracle (2023). *MySQL Feature Comparison*, Oracle.
  30. Oracle (2023), *Why MySQL? Performance, Scalability & Flexibility*, Oracle Corporation.
  31. Pandas Documentation (2023), *Pandas User Guide*, Version 2.1, Pandas Project
  32. Python Software Foundation – PSF (2023), *About Python – Use cases and adoption*, PSF.
  33. React Router (2023), *React Router Documentation*, Remix Software.

34. Scikit-learn (2023), *Scikit-learn: Machine Learning in Python – Documentation*, Version 1.3.
35. Seaborn (2023), *Seaborn Statistical Data Visualization Library*, Version 0.12, PyData.
36. Smith, J., Brown, T., & Lee, M. (2020), *Machine learning models for diabetes classification using PIMA dataset*, *Journal of Medical Informatics*, 27(3), pp. 112–120.
37. Stack Overflow (2023), *Stack Overflow Developer Survey 2023 – Key Results*, Stack Overflow.
38. TailwindCSS (2023), *Tailwind CSS Documentation*, Tailwind Labs Inc.
39. The Lancet Digital Health (2021), “Machine learning models for diabetes risk prediction”, *The Lancet Digital Health*.
40. World Health Organization – WHO (2016), *Global Report on Diabetes*, Geneva.
41. World Health Organization – WHO (2020), *Prevention of diabetes: evidence and impact*, WHO, Geneva.
42. World Health Organization (WHO). *Diabetes – Key facts*, 2021.
43. World Health Organization – WHO (2023), *Diabetes key facts – Updated 6 April 2023*, WHO, Geneva.