# Capstone Project 2: Heart Attack Risk Prediction - Data Modeling Report

Student Name: Samuel Olumide Adebimpe

## Step 1: Dataset Summary

The dataset used in this project is titled "Heart Attack Risk Prediction (Cleaned Dataset)", sourced from Kaggle. It contains structured data in CSV format, organized into rows (each representing a patient) and columns (each representing a medical or lifestyle attribute.

- Type: Structured, quantitative and categorical.

- Format: CSV

- Source: Kaggle (Original data from UCI ML Repository)

- Download Method: Manual (via Opera browser)

- Dataset URL: https://www.kaggle.com/datasets/alikalwar/heart-attack-risk-prediction-cleaned-dataset

**Attributes Include:**

- Vitals (e.g., SystolicBP, DiastolicBP, HeartRate, BMI)

- Lab Results (e.g., BloodSugar, Cholesterol, Triglycerides, Troponin, CK-MB)

- Lifestyle factors (e.g., Smoking, Alcohol, Diet, Exercise)

- Risk assessments (e.g., StressLevel, RiskBinary)

- Medical history (e.g., Diabetes, FamilyHistory)

The dataset is suitable for building predictive models and supports structured storage for analytics.

## Step 2: Conceptual Data Model

The conceptual model identifies eight core entities with their general relationships:

**Entities:**

- Patient

- VitalSigns

- LabResults

- Lifestyle

- MedicalHistory

- RiskAssessment

- PredictionLog

- DietaryHabits

**Relationships & Justification:**

- Patient (1) → (M) VitalSigns: One patient can have multiple blood pressure, heart rate, and BMI readings over time.

- Patient (1) → (M) LabResults: Lab tests such as cholesterol or troponin can be repeated multiple times.

- Patient (1) → (M) Lifestyle: Lifestyle data may be updated or recorded at different times (e.g., change in diet or exercise habits).

- Patient (1) → (1) MedicalHistory: A patient’s medical history is generally a consolidated and relatively static record (e.g., diabetes, family history).

- Patient (1) → (M) RiskAssessment: Stress levels and heart attack risk may be reassessed regularly, producing multiple entries.

- Patient (1) → (M) PredictionLog: Each model prediction made for a patient is logged with date, risk, and method used.

- Patient (1) → (M) DietaryHabits: Dietary logs may be captured repeatedly to monitor changes or improvements.

The conceptual ERD uses crow’s foot notation to represent 1:M and 1:1 relationships clearly. This model is ideal for healthcare systems where patient-centered records are maintained over time.

## Step 3: Logical Data Model (3NF)

The logical model is fully normalized to 3rd Normal Form (3NF):

- All attributes are atomic.

- No transitive dependencies exist.

- Each table has a primary key (PK) and relevant foreign key (FK) relationships.

- One-to-many (1:M) and one-to-one (1:1) relationships are preserved.

Each entity includes appropriate fields:

- VitalSigns: SystolicBP, DiastolicBP, HeartRate, BMI, each with corresponding level columns.

- LabResults: Cholesterol, BloodSugar, CK\_MB, Troponin, Triglycerides, with level columns.

- Lifestyle: Smoking, AlcoholConsumption, Obesity, Diet, Sleep and Sedentary hours.

- MedicalHistory: Diabetes, FamilyHistory, PreviousHeartProblems, MedicationUse.

- RiskAssessment: StressLevel, HeartAttackRiskBinary, RiskText.

- PredictionLog: Date, PredictedRisk, ModelUsed.

- DietaryHabits: DietType, DietScore.

Refer to the attached Logical ERD PDF for full schema.

## Step 4: Physical Data Model & SQL Schema

The physical model translates the logical design into SQLite implementation:

- Data types defined: INTEGER, DECIMAL, TEXT, BOOLEAN, DATETIME

- Foreign key constraints ensure data integrity.

- Tables were created successfully using DBeaver (SQLite)

SQL Scripts Include:

- CREATE TABLE statements for all 8 entities

- Appropriate primary and foreign keys

- Unique constraint for 1:1 MedicalHistory table

Example Snippet:

CREATE TABLE Patient (

PatientID INTEGER PRIMARY KEY,

Age INTEGER,

Gender TEXT,

Income DECIMAL

);

**Refer to attached Physical ERD PDF and SQL Scripts.**

## Step 5: SQL Queries & Results

**✅ Easy Query 1: Count total patients**

SELECT COUNT(\*) AS TotalPatients FROM Patient;

**✅ Easy Query 2: Average BMI**

SELECT AVG(BMI) AS AverageBMI FROM VitalSigns;

**✅ Medium Query 1: Cholesterol level and stress**

SELECT Patient.PatientID, LabResults.Cholesterol\_Level, RiskAssessment.StressLevel\_Category

FROM Patient

JOIN LabResults ON Patient.PatientID = LabResults.PatientID

JOIN RiskAssessment ON Patient.PatientID = RiskAssessment.PatientID;

**✅ Medium Query 2: Obese patients at risk**

SELECT COUNT(\*) AS ObeseHighRiskCount

FROM Lifestyle

JOIN RiskAssessment ON Lifestyle.PatientID = RiskAssessment.PatientID

WHERE Obesity = 1 AND HeartAttackRiskBinary = 1;

**✅ Difficult Query: Join 4 tables**

SELECT Patient.PatientID, VitalSigns.BMI, RiskAssessment.HeartAttackRiskBinary,

PredictionLog.ModelUsed, PredictionLog.PredictedRisk

FROM Patient

JOIN VitalSigns ON Patient.PatientID = VitalSigns.PatientID

JOIN RiskAssessment ON Patient.PatientID = RiskAssessment.PatientID

JOIN PredictionLog ON Patient.PatientID = PredictionLog.PatientID

WHERE VitalSigns.BMI\_Level = 'High'

AND RiskAssessment.HeartAttackRiskBinary = 1

AND PredictionLog.PredictedRisk > 0.7

ORDER BY PredictionLog.PredictedRisk DESC;