VISHWA – Vital Information System for Health Wellness and Awareness

(VISHWA)

Objective

We aim to improve global health care by creating a modern, data-based system that tracks diseases, makes the best use of health resources, and provides real-time health information. **VISHWA** gives doctors, health officials, and researchers the useful information they need to act quickly, focus on prevention, and protect public health worldwide.

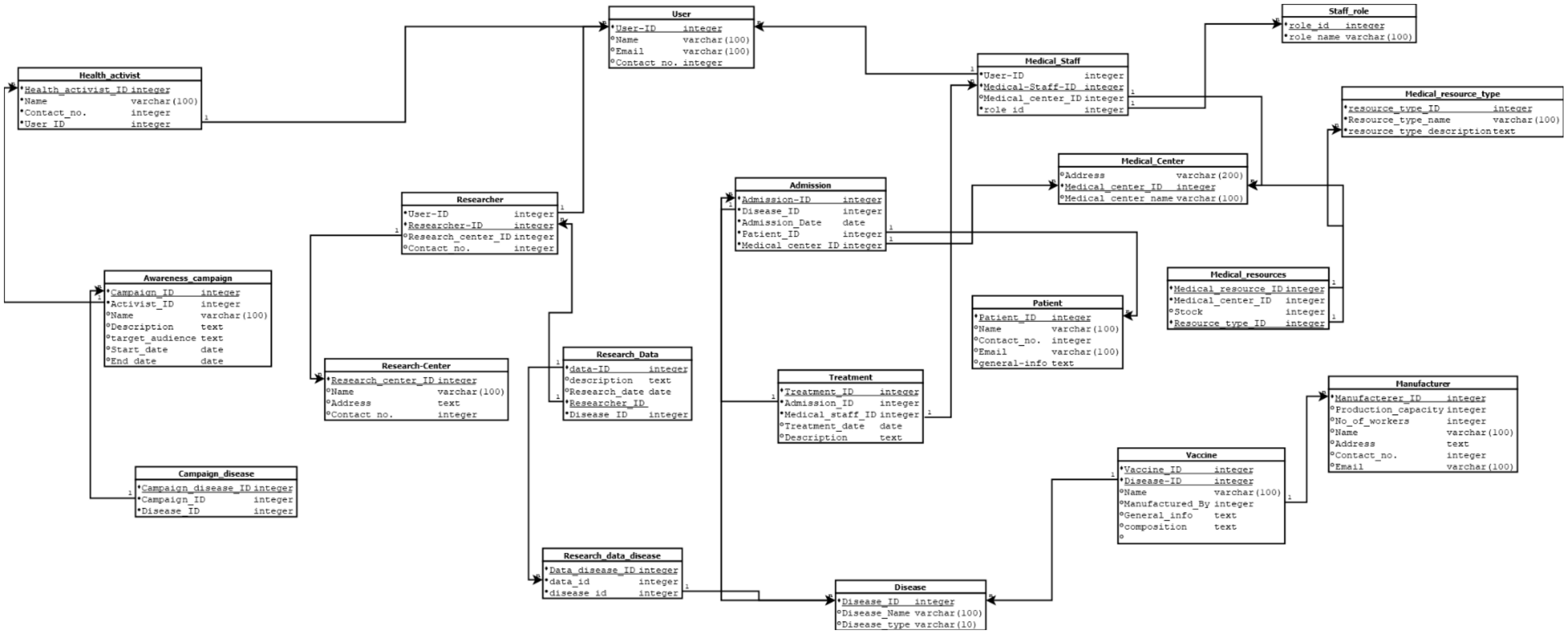
Lab Group: 6

Project Group No: 2

Group Members

- **Akshat Bhatt (Group Representative) -** 202301460
- **Maheriya Harsh Prakashbhai -** 202301470
- **Nilesh Mori -** 202301473

Relational Schema :



Functional Dependencies and Minimal FD Set:

- **USER**
- PK : User_ID
    - User_ID → Name
    - User_ID → Email
    - User_ID → Contact_No

- **RESEARCHER**
- PK : Researcher_ID
    - Researcher_ID → User_ID
    - Researcher_ID → Research_Center_ID

- **RESEARCH_CENTER**
- PK : Research_Center_ID
    - Research_Center_ID → Name
    - Research_Center_ID → Address

- **RESEARCH_DATA**
- PK : Data_ID
    - Data_ID → Researcher_ID
    - Data_ID → Description

  - Data_ID → Research Date
  - Data_ID → Disease ID

- **DISEASE**
- PK : Disease_ID
  - Disease_ID → Disease_Name
  - Disease_ID → Disease_type

- **RESEARCH_DATA_DISEASE**
- PK : Data_Disease_ID
  - Data_Disease_ID → Data_ID
  - Data_Disease_ID → Disease_ID

- **HEALTH_ACTIVIST**
- PK : Health_activist_ID
  - Health_activist_ID → User_ID
  - Health_activist_ID → Name
  - Health_activist_ID → Contact_no

- **AWARENESS_CAMPAIGN**
- PK : Campaign_ID
  - Campaign_ID → Activist_ID
  - Campaign_ID → Name

  - Campaign_ID → Description
  - Campaign_ID → Target_Audience
  - Campaign_ID → Start_Date
  - Campaign_ID → End_Date

- **CAMPAIGN_DISEASE**
- PK : Campaign_Disease_ID
  - Campaign_Disease_ID → Campaign_ID
  - Campaign_Disease_ID → Disease_ID

- **MEDICAL_STAFF**
- PK : Medical_Staff_ID
  - Medical_Staff_ID → User_ID
  - Medical_Staff_ID → Medical_Center_ID
  - Medical_Staff_ID → Role_ID
- **STAFF_ROLE**
- PK : Role_ID
  - Role_ID → Role_Name

- **MEDICAL_CENTER**
- PK : Medical_Center_ID
  - Medical_Center_ID → Address
  - Medical_Center_ID → Medical_Center_Name

- **MEDICAL_RESOURCE**
- PK : Medical_Resource_ID
    - Medical_Resource_ID → Medical_Center_ID
    - Medical_Resource_ID → Resource_type_ID
    - Medical_Resource_ID → Stock

- **RESOURCE_TYPE**
- PK : Resource_Type_ID
    - Resource_Type_ID → Resource_Type_Name
    - Resource_Type_ID→ Resource_type_Description

- **PATIENT**
- PK : Patient_ID
    - Patient_ID → Name
    - Patient_ID → Email
    - Patient_ID → Contact No
    - Patient_ID → General_Info

- **ADMISSION**
- PK : Admission_ID
    - Admission_ID → Patient_ID
    - Admission_ID → Disease_ID
    - Admission_ID → Admission_Date
    - Admission_ID → Medical_Center_ID

- **TREATMENT**
- PK : Treatment_ID
    - Treatment_ID → Admission_ID
    - Treatment_ID → Medical_Staff_ID
    - Treatment_ID → Treatment_Date
    - Treatment_ID → Description

- **VACCINE**
- PK : Vaccine_ID
    - Vaccine_ID → Disease_ID
    - Vaccine_ID → Name
    - Vaccine_ID → Manfacturer_ID
    - Vaccine_ID → Composition
    - Vaccine_ID → General_info

- **MANUFACTURER**
- PK : Manufacturer_ID
    - Manufacturer_ID → Name
    - Manufacturer_ID → Address
    - Manufacturer_ID → Contact_No
    - Manufacturer_ID → Email
    - Manufacturer_ID → Production_capacity

Proof Of BCNF:

For every relation:

- **The only nontrivial FD is PK→other attributes\text{PK} \to \text{other attributes}PK→other attributes.**
  - The left-hand side (LHS) of every FD is the primary key.
- **By definition, the primary key is a superkey.**
  - This means that in each case, the determinant of the FD is a superkey.
- **Since every FD satisfies the condition that the LHS is a superkey, each relation meets the criteria for BCNF.**

- **To ensure the schema adhered strictly to BCNF Campaign_disease table was added to avoid storing multi-valued Disease_ID in Awareness_campaign.**

- **Staff_role was introduced to manage the possibility of multiple roles per staff member, which would violate BCNF if stored directly in Medical_Staff.**

- **These structural additions prevent partial and transitive dependencies, ensuring data integrity and minimal redundancy across the database schema.**

DDL Script :

```
------------------------------------------------------------------------------------------------------------------------

CREATE TABLE App_User (
   User_ID INTEGER PRIMARY KEY,
   Name VARCHAR(100),
   Email VARCHAR(100),
   Contact_no INTEGER
);

CREATE TABLE Health_activist (
   Health_activist_ID INTEGER PRIMARY KEY,
   Name VARCHAR(100),
   Contact_no INTEGER,
   User_ID INTEGER,
   FOREIGN KEY (User_ID) REFERENCES App_User(User_ID)
);

CREATE TABLE Staff_role (
   role_id INTEGER PRIMARY KEY,
   role_name VARCHAR(100)
);

CREATE TABLE Medical_Center (
   Medical_center_ID INTEGER PRIMARY KEY,
   Medical_center_name VARCHAR(100),
   Address VARCHAR(200)
);

CREATE TABLE Medical_Staff (
   User_ID INTEGER,
   Medical_Staff_ID INTEGER PRIMARY KEY,
   Medical_center_ID INTEGER,
   role_id INTEGER,
   FOREIGN KEY (User_ID) REFERENCES App_User(User_ID),
   FOREIGN KEY (Medical_center_ID) REFERENCES Medical_Center(Medical_center_ID),
   FOREIGN KEY (role_id) REFERENCES Staff_role(role_id)
);

CREATE TABLE Patient (
   Patient_ID INTEGER PRIMARY KEY,
   Name VARCHAR(100),
   Contact_no INTEGER,
   Gender VARCHAR(10),
   General_info TEXT
);

CREATE TABLE Disease (
   Disease_ID INTEGER PRIMARY KEY,
   Disease_Name VARCHAR(100),
   General_info TEXT
);

CREATE TABLE Admission (
   Admission_ID INTEGER PRIMARY KEY,
   Disease_ID INTEGER,
   Admission_date DATE,
   Patient_ID INTEGER,
   Medical_center_ID INTEGER,
   FOREIGN KEY (Disease_ID) REFERENCES Disease(Disease_ID),
   FOREIGN KEY (Patient_ID) REFERENCES Patient(Patient_ID),
   FOREIGN KEY (Medical_center_ID) REFERENCES Medical_Center(Medical_center_ID)
);

CREATE TABLE Treatment (
   Treatment_ID INTEGER PRIMARY KEY,
   Admission_ID INTEGER,
   Medical_staff_ID INTEGER,
   Treatment_start_date DATE,
   Treatment_end_date DATE,
   Description TEXT,
   FOREIGN KEY (Admission_ID) REFERENCES Admission(Admission_ID),
   FOREIGN KEY (Medical_staff_ID) REFERENCES Medical_Staff(Medical_Staff_ID)
);

CREATE TABLE Medical_resource_type (
   Resource_type_ID INTEGER PRIMARY KEY,
   Resource_type_name VARCHAR(100),
   Resource_type_description TEXT
);

CREATE TABLE Medical_resources (
   Medical_resource_ID INTEGER PRIMARY KEY,
   Medical_center_ID INTEGER,
   Stock INTEGER,
   Resource_type_ID INTEGER,
   FOREIGN KEY (Medical_center_ID) REFERENCES Medical_Center(Medical_center_ID),
   FOREIGN KEY (Resource_type_ID) REFERENCES
Medical_resource_type(Resource_type_ID)
);

CREATE TABLE Manufacturer (
   Manufacturer_ID INTEGER PRIMARY KEY,
   Production_capacity INTEGER,
   No_of_workers INTEGER,
   Name VARCHAR(100),
   Address TEXT,
   Email VARCHAR(100)
);

CREATE TABLE Vaccine (
   Vaccine_ID INTEGER PRIMARY KEY,
   Disease_ID INTEGER,
   Name VARCHAR(100),
   Manufactured_by INTEGER,
   Medical_info TEXT,
   Composition TEXT,
   FOREIGN KEY (Disease_ID) REFERENCES Disease(Disease_ID),
   FOREIGN KEY (Manufactured_by) REFERENCES Manufacturer(Manufacturer_ID)
);

CREATE TABLE Research_Center (
   Research_center_ID INTEGER PRIMARY KEY,
   Name VARCHAR(100),
   Address TEXT
);

CREATE TABLE Researcher (
   User_ID INTEGER,
   Researcher_ID INTEGER PRIMARY KEY,
   Research_center_ID INTEGER,
   FOREIGN KEY (User_ID) REFERENCES App_User(User_ID),
   FOREIGN KEY (Research_center_ID) REFERENCES
Research_Center(Research_center_ID)
);

CREATE TABLE Research_Data (
   Data_ID INTEGER PRIMARY KEY,
   Description TEXT,
   Research_date DATE,
   Researcher_ID INTEGER,
   Disease_ID INTEGER,
   FOREIGN KEY (Researcher_ID) REFERENCES Researcher(Researcher_ID),
   FOREIGN KEY (Disease_ID) REFERENCES Disease(Disease_ID)
);

CREATE TABLE Research_data_disease (
   Data_disease_ID INTEGER PRIMARY KEY,
   Data_ID INTEGER,
   Disease_ID INTEGER,
   FOREIGN KEY (Data_ID) REFERENCES Research_Data(Data_ID),
   FOREIGN KEY (Disease_ID) REFERENCES Disease(Disease_ID)
);

CREATE TABLE Awareness_campaign (
   Campaign_ID INTEGER PRIMARY KEY,
   Health_ID INTEGER,
   Name VARCHAR(100),
   Description TEXT,
   Target_audience TEXT,
   Start_date DATE,
   End_date DATE,
   FOREIGN KEY (Health_ID) REFERENCES Health_activist(Health_activist_ID)
);

CREATE TABLE Campaign_disease (
   Campaign_disease_ID INTEGER PRIMARY KEY,
   Campaign_ID INTEGER,
   Disease_ID INTEGER,
   FOREIGN KEY (Campaign_ID) REFERENCES Awareness_campaign(Campaign_ID),
   FOREIGN KEY (Disease_ID) REFERENCES Disease(Disease_ID)
);
```

THANK YOU
