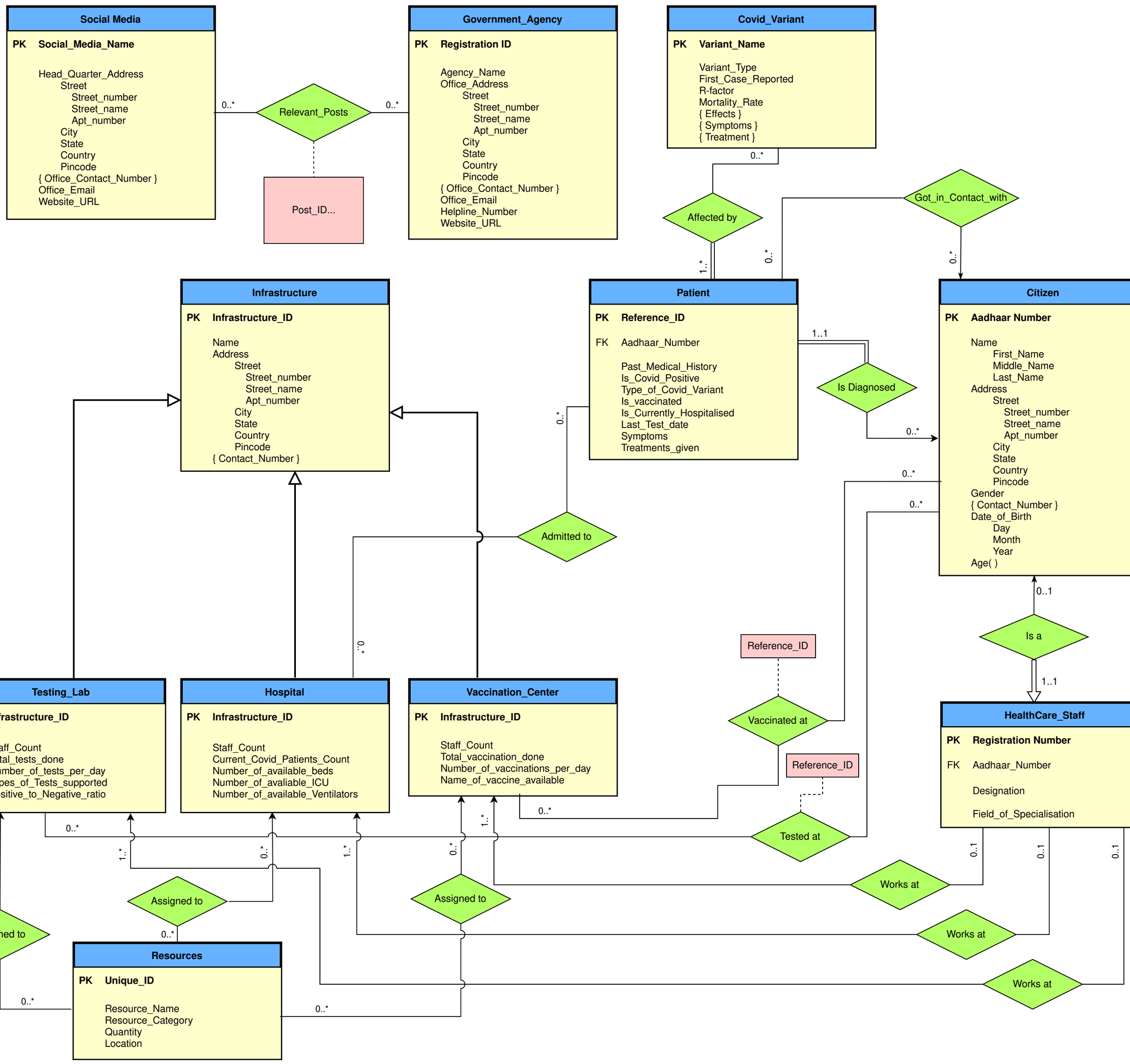
Database Management System Lab

Assignment - 1
E-R Diagram

Pranav Mehrotra, 20CS10085



Text is not SVG - cannot display

Entities and their attributes:

a) Government_Agency:

Government_Agency (<u>Registration_id</u>, Agency_Name, Street, Street_number, Street_name, Apt_number, City, State, Country, Pincode, Office_Email, Helpline_Number, Website_URL)

Primary-key: Registration_id

MultiValued Attribute:
Govern_Office_Contact=(<u>Registration_id</u>, Office_Contact_Number)

Along with the agency's address, the entity also maintains the name of the government agency, its address, office phone number, helpline number, and website address. Every time a new agency is added to the database, a unique primary key called **<u>Registration_id</u>** is created.

b) Citizen:

Citizen (<u>Aadhaar_number</u>, First_Name, Middle_Name, Last_Name, Street, Street_number, Street_name, Apt_number, City, State, Country, Pincode, Gender, Day, Month, Year)

Primary-key: Aadhaar_number

MultiValued Attribute:
Citizen_Contact=(<u>Aadhaar_number</u>, Contact_Number)
The Citizen entity would store the name, contact, date of birth, gender and other information related to their address. **Aadhaar number** would be the primary key and would be unique for all entries.

c) HealthCare_Staff:
HealthCare_Staff (<u>Registration_Number</u>, Aadhaar_Number, designation, specialization, qualification)

Primary-key: Registration_Number
Foreign-key: Aadhaar_Number
The entity would store the Registration number of the Health Care Staff which would act as a Primary key to the table. The **Aadhaar number** would act as the foreign key for the entity which connects it to the Citizen entity. Apart from that Field of Specialisation of the doctor and their designation say, doctor, nurses or surgeons are stored.

d) Patient:
Patient (Reference_ID, Aadhaar_number, Past_Medical_History, Is_Covid_Positive, Type_of_Covid_Variant, Is_vaccinated, Is_Currently_Hospitalised, Last_Test_date, Symptoms, Treatments_given)

Primary-key: Reference_ID
Foreign-key: Aadhaar_number (to Citizen Entity)

The patient's Aadhaar number, which serves as a foreign key to the citizen database, would be stored by the organisation. A brief medical history that details their past health state, their vaccination status, previous illnesses and treatments, and if they are hospitalised or in self-isolation is kept. In addition, records of their symptoms and the medical care they received are kept. Every time a citizen is given a diagnosis of COVID, a special reference id is generated for that person. The system would generate the reference id, which would serve as the database's primary key.

This table would be populated with the information provided by hospitals about the patient or the patient themself (**via self-reporting or social media**).

e) For medical infrastructure:
   i) Infrastructure:

Infrastructure (Infrastructure_ID, Name, Street, Street_number, Street_name, Apt_number, City, State, Country, Pincode

Primary Key: Infrastructure_ID

MultiValued Attribute:
Infrastructure_Contact=(Infrastructure_ID, Contact_Number)

The entity would keep a record of the name, address, and phone number of the medical infrastructure, such as a hospital, immunisation clinic, or testing lab. Every entry in the table would be given a distinct **Infrastructure ID** (primary key).

ii) Specialized class: Hospital:

Hospital (Infrastructure_ID, Staff_Count, Current_Covid_Patients_Count, Number_of_available_beds, Number_of_avaliable_ICU, Number_of_available_Ventilators)

Primary Key: Infrastructure_ID

The number of staff members, the number of active patients, and the number of available beds, ICU beds, and ventilators would all be stored in the entity. It is preferable to record the number of ICUs, beds, and ventilators as properties within the hospital class itself because different users would frequently request information on these topics. The Infrastructure table, which would have some fundamental data about the hospital, would function as a foreign key with Infrastructure ID as its primary key.

iii) Specialized class: Testing_lab:

Testing_Lab (Infrastructure_ID, Staff_Count, Total_tests_done, Number_of_tests_per_day, Types_of_Tests_supported, Positive_to_Negative_ratio)

Primary Key: Infrastructure_ID

The entity store the **Infrastructure_ID** inherited from the parent class that would act as the foreign key to the Infrstructure table. Along with that, the type of testing supported say PCR, Rapid Antigen; Positive Reports to Negative Reports ratio, Total number of Tests done, Number of Testing done per day, number of staff.

iv) Specialized class: Vaccination_Center:

Vaccination_Center (Infrastructure_ID, Staff_Count, Total_vaccination_done, Number_of_vaccinations_per_day, Name_of_vaccine_available)

Primary Key: Infrastructure_ID

The entity inherits the **Infrastructure_ID** from the parent that would act as the foreign key to the Infrastructure table. Along with that number of staff available, the average number of people vaccinated per day, the total number of vaccinations done till date, and the Names of vaccines provided are stored.

f) Covid_Variant:

Covid_Variant (Variant_Name, Variant_Type, First_Case_Reported, R-factor, Mortality_Rate, Effects, Symptoms, Treatment)

Primary-key: Variant_Name

MultiValued Attributes:
Variant_Effects=(Variant_Name, Effects)
Variant_Symptoms=(Variant_Name, Symptoms)
Variant_Treatments=(Variant_Name, Treatments)

The variant entity store the name of different Covid variants like B.1.1.529 (Omnicron), it acts as a **unique variant key** (primary key) to identify every variant uniquely. The entity also stores the R-factor of the variant that determines the spreadability of the virus, mortality_rate of the variant along with details about the date of the First case reported of the virus. Effects, Symptoms and Treatments would be multi-valued attributes to save information about the commonly

observed attributes of the variant. It will also help heath professionals to get an idea of how to identify and treat certain variants of Covid.

g) Resources:

Resources (Unique_ID, Resource_Name, Resource_Category, Quantity, Location)

Primary Key: Unique_ID

The entity would store the resource's name, which can be Medicines, Injections, Ambulances, Beds, ICUs, ventilators, and other medical supplies, and their available quantities at different locations. Every resource entered would have a unique identifier named **Unique_ID**(primary key).

h) Social_Media:

Social_Media(Social_Media_Name, Street, Street_number, Street_name, Apt_number, City, State, Country, Pincode, Office_Email, Website_URL)

Primary-key: Social_Media_Name

MultiValued Attribute:
Social_Office_Contact=(Social_Media_Name, Office_Contact_Number)

The social media entity would store the registered name of the social media platform, its Head Quarter address, and the contact details of their office. This table would contain all social media platforms the government might use to gather and spread vital information.

Hospitals, Social media, and self-reporting are all potential sources for Covid Related Information.
Our understanding of the current state of immunisation, the transmission of the virus among individuals from various age groups and geographic locations, frequent viral symptoms, and the quantity of resources currently accessible to the government will be

aided by the information contained in these tables. When retrieved, these bits of data can be utilised to halt the spread of rumours or false information on social media.

To track symptoms and Treatments of various Variants, Healthcare professionals can query in the database and get the relevant information from the Covid_Variant table.

Relational tables:

a) citizen-patient (is_diagnosed):

A **citizen** who has been diagnosed with COVID is a **patient**, also one citizen can be affected by COVID multiple times, making the **citizen-patient** relationship a **one-to-many** relationship. Every patient will have a corresponding item in the citizen database, meaning that the patient has **total participation**; however, not every citizen would be a patient.

is_diagnosed (aadhaar_num, reference_id)
For all patients, the relation is_diagnosed would contain tuples of the form (citizen's aadhaar number and patient reference id) (citizens who are diagnosed with covid). This relationship table would provide the patient's Aadhaar number, which can be utilised to retrieve information from the citizen table, while the reference id would assist us in retrieving information from the patient table.

b) citizen-patient (in_contact):

To keep track of which patients were all in close proximity to one another, in contact relation would be utilised (contact tracing). It would be a **many-to-many** relationship, meaning that numerous patients can get in touch of the same citizen, and numerous citizens might get in touch with the same patient. There may be citizens or patients who haven't interacted with any other parties, meaning that **none of the entities has total participation**.

Got_in_contact_with(aadhaar_num, reference_id)
This relation table stores the aadhaar number of all citizens who have been in contact with the patient referred by reference id. This would be used for contact tracing.

c) Patient-hospital (admitted_at):

A patient is admitted at that hospital. This relation would be **many to many,** indicating a patient might be hospitalised several times at different hospitals. **None of the two entities has total participation** in the relation.

admitted_to(reference_id, infrastructure_id)

Which patient received care at which hospital is found in this table. The infrastructure_id would assist us in accessing the information about the hospital, and reference_id would assist us in doing the same for the patient.

d) Citizen-vaccination_center (Vaccinated_at):

If a citizen received a dose at a vaccination center, they are related. Considering that a citizen can receive several vaccinations at various vaccination facilities and multiple citizens can receive vaccinations at the same facility. As a result, **none of the entities is in total participation** in this **many-to-many relationship**.

Vaccinated_at(Aadhaar_number, Reference_id, Infrastructure_id)

Each vaccine would produce a distinct Reference_ID (analogous to the reference id present in the Cowin vaccination). Aadhaar numbers of citizens and identification from immunisation facilities would be kept. The related table summarises the specifics of each citizen's immunisation. The only field that is unique to all records is reference_id.

e) Citizen-testing_lab (Tested_at):
The two entities are related if a citizen has had a covid test performed at a testing lab. Multiple people can be tested at the same testing facility and a citizen can take multiple tests at various locations. As a result, **none of the entities is totally participating** in this **many-to-many relationship**.

Tested_at(aadhaar_num, reference_id, infra_id)

A distinct reference_id would be generated for each test. Both the center's ID and the citizens' Aadhaar numbers would be stored. This relationship table summarises each citizen's testing background.

f) Patient-variant (affected_by):
A patient would be connected to the variant with which the variant table has associated him. The relation would be **many-to-many** because a patient could be diagnosed with several variants and a single mutation could affect many people. The patient entity has **total participation** in the connection because each patient would be impacted by atleast one of the variants.

affected_by(reference_id, variant_name)

The table would have the variant name a patient is affected by. The two attributes of this table can help us in retrieving information about every patient and the variant they have been affected by.

g) Healthcare_staff-Hospital (works_at):
There would be a connection between the hospital and the health care staff where they work. Many healthcare staff might be stationed at the same hospital; therefore, there would be a **many-to-one** relationship, and **no entity would have total participation**. A health officer is also limited to working in a single hospital.

works_at(registration_id, infrastructure_id)

h) Healthcare_staff-vaccination_center(works_at):
There would be a connection between the vaccination center and the health care staff where they work. Many healthcare staff might be stationed at the same vaccination center; therefore, there would be a **many-to-one** relationship, and **no entity would have total participation**. A health officer is also limited to working in a single vaccination center.

works_at(registration_id, infrastructure_id)

i) Healthcare_staff-testing_lab(works_at):

There would be a connection between the testing lab and the health care staff where they work. Many healthcare staff might be stationed at the same testing lab; therefore, there would be a **many-to-one** relationship, and **no entity would have total participation**. A health officer is also limited to working in a single testing lab.

works_at(registration_id, infrastructure_id)

j) Resources-hospital (Assigned_to):
This relationship associates a hospital with a medical resource, i.e. inventory of hospitals. This relation would store information about which resources belong to which hospitals and the quantity of available resources. The government can look at the figures for Ambulances, beds, ICUs, ventilators, and medicines and make decisions about future readiness. This relationship would be a **many-to-one** as many resources can be allocated to the same hospital, but the same resource cannot be allocated to two different hospitals. **No entity would have total participation**.

assigned_to(<u>resource_id</u>, Infrastructure_id)

Each record's resource_id would be different. This relational table will assist us in locating the resource's original location as indicated by resource id. The information about the hospital would then be extracted using the infrastructure_id, and information about the resource availability within the hospital would be extracted using the resource_id.

k) Resources-vaccination_center (Assigned_to):
This relationship associates a vaccination center with a medical resource, i.e. inventory of vaccination centers. This relation would store information about which resources belong to which vaccination centers and the quantity of available resources. The government can look at the figures for different types of vaccines available and make decisions about future readiness. This relationship would be a **many-to-one** as many resources can be allocated to the same vaccination center, but the same resource cannot be allocated to two different vaccination centers. **No entity would have total participation**.

assigned_to(<u>resource_id</u>, Infrastructure_id)

Each record's resource_id would be different. This relational table will assist us in locating the resource's original location as indicated by resource id. The information about the vaccination center would then be extracted using the infrastructure_id, and information about the resource availability within the vaccination center would be extracted using the resource_id.

l) Resources-testing_lab (Assigned_to):
This relationship associates a testing lab with a medical resource, i.e. inventory of testing labs. This relation would store information about which resources belong to which testing labs and the quantity of available resources. The government can look at the figures for different tests available and make decisions about future readiness. This relationship would be a **many-to-one** as many resources can be allocated to the same testing lab, but the same resource cannot be allocated to two different testing labs. **No entity would have total participation**.

assigned_to(<u>resource_id</u>, Infrastructure_id)

Each record's resource_id would be different. This relational table will assist us in locating the resource's original location as indicated by resource id. The information about the testing lab would then be extracted using the infrastructure_id, and information about the resource availability within the testing lab would be extracted using the resource_id.

m) Citizen-health_official (is_a):
A citizen will be related to a health official if they are a health_official, i.e., this relationship is a one-to-one map with all health officials being a citizen too, which means total participation from a health_official entity.

is_a (<u>aadhaar_num</u>, registration_id)

The relationship will exist if the citizen is a healthcare staff. The registration_id would give us information about the health official's information; the Aadhaar number would give us general information about them.

n) Social_media-Government_Agency (Relevant_posts):

This relationship would map a social media company with government agencies, with the relation being a post containing covid info posted by the government. Only government agency posts would be stored as they are a source of authentic information.

Relevant_posts (<u>Registration_id</u>, <u>Social_media_Name</u>, <u>Post_ID</u>, Post_Description, Post_Account, Post_Location)

This connection would serve as an information hub where critical posts pertaining to Covid-19 would be kept to give government agencies accurate information and to prevent the spread of rumours and misinformation. The posts may cover topics such as updates to Covid-19 or any local news about a community or society.