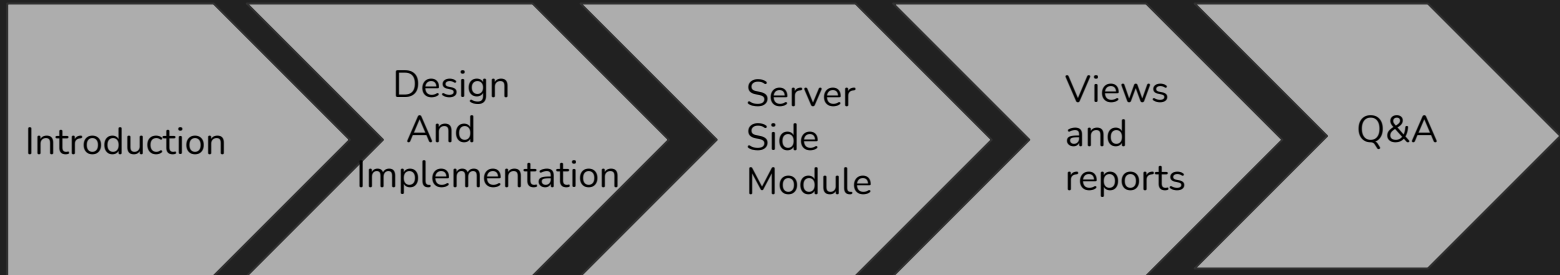


Contact Tracing



Road Map



Motivation

- COVID-19 seems to be spreading easily and sustainably in the community in many affected geographic areas.

Solutions ?

- - Stay at home
- - Contact Tracing

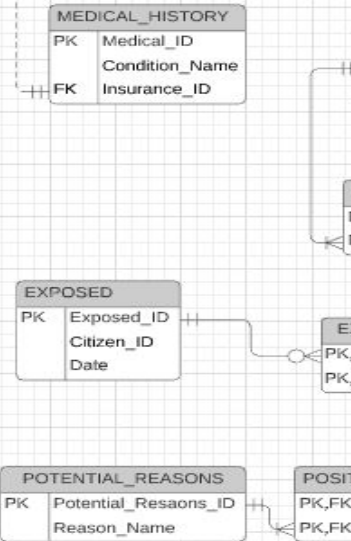
What is Contact Tracing?

Process of tracking down an individual who has had an infection and the people that person has potentially been in contact with.

Goals of our db design

- Track citizen's location history
- Track positive cases along with their symptoms and potential reasons
- Manage details of hospitals and quarantine centers.
- To store Active , Death and Recovered cases
- To identify Critical , Moderate and Safe Zones in a state.

CITIZEN ID 18 LOCATION HISTORY



ID	DateTime	GPS(Latitude,Longitude)
1	2020-02-03,09:00:00	Latitude: 37.276715520647336 Longitude: -121.83621938659853
2	2020-02-03,13:00:00	Latitude: 32.276715520647336 Longitude: -124.83621938659853
3	2020-02-03,15:30:00	Latitude: 40.276715520647336 Longitude: -123.83621938659853
4	2020-02-03,19:30:00	Latitude: 20.276715520647336 Longitude: -110.83621938659853
5	2020-02-04,14:30:00	Latitude: 19.276715520647336 Longitude: -108.83621938659853
6	2020-02-05,8:30:00	Latitude: 15.276715520647336 Longitude: -124.83621938659853
7	2020-02-05,17:40:00	Latitude: 22.276715520647336 Longitude: -150.83621938659853
8	2020-02-06,12:50:00	Latitude: 18.276715520647336 Longitude: -140.83621938659853
9	2020-02-07,12:50:00	Latitude: 12.276715520647336 Longitude: -160.83621938659853
9	2020-02-07,20:20:00	Latitude: 19.4576715520647336 Longitude: -160.66621938659853
10	2020-02-08,22:20:00	Latitude: 30.4576715520647336 Longitude: -141.66621938659853
11	2020-02-08,13:20:00	Latitude: 20.4576715520647336 Longitude: -145.66621938659853
12	2020-02-09,14:55:00	Latitude: 21.4576715520647336 Longitude: -146.66621938659853
13	2020-02-10,16:55:00	Latitude: 22.4576715520647336 Longitude: -177.66621938659853
14	2020-02-11,15:35:00	Latitude: 28.4576715520647336 Longitude: -140.66621938659853
15	2020-02-12,16:35:00	Latitude: 20.4576715520647336 Longitude: -148.66621938659853
16	2020-02-12,17:35:00	Latitude: 20.4576715520647336 Longitude: -148.66621938659853
17	2020-02-13,10:35:00	Latitude: 22.4576715520647336 Longitude: -138.66621938659853
18	2020-02-14,23:59:00	Latitude: 100.4576715520647336 Longitude: -200.66621938659853
19	2020-02-15,16:35:00	Latitude: 20.4576715520647336 Longitude: -148.66621938659853
20	2020-02-16,13:15:00	Latitude: 30.4576715520647336 Longitude: -145.66621938659853
21	2020-02-16,22:35:00	Latitude: 9.4576715520647336 Longitude: -110.66621938659853
22	2020-02-17,16:35:00	Latitude: 20.4576715520647336 Longitude: -148.66621938659853
23	2020-02-17,16:35:00	Latitude: 20.4576715520647336 Longitude: -148.66621938659853
24	2020-02-17,06:30:00	Latitude: 45.4576715520647336 Longitude: -112.66621938659853

Table DDL

```
CREATE TABLE Citizen
(
  CitizenID INT IDENTITY NOT NULL PRIMARY KEY,
  FirstName Varchar(20) NOT NULL,
  LastName Varchar(20) NOT NULL,
  DOB Date NOT NULL,
  Gender Char(10) NOT NULL,
  GPS Varchar(40) NOT NULL
  Age as dbo.datediff(DOB)|
);
```

```
CREATE TABLE POSITIVE
(
  PatientID varchar(5) NOT NULL PRIMARY KEY ,
  TestedID varchar(20) NOT NULL REFERENCES Tested(testedID),
  HospitalID varchar(5) NOT NULL REFERENCES Hospital(HospitalID),
  Severity_Of_Condition varchar(40) NOT NULL,
  Patient_Status varchar(40) NOT NULL,
  Positive_Date datetime DEFAULT Current_Timestamp
);
```

```
CREATE TABLE Insurance
(
  InsuranceID Varchar(20) NOT NULL PRIMARY KEY,
  CitizenID INT NOT NULL REFERENCES Citizen(CitizenID),
  Company_Name Varchar(20),
  Validity Date|
);
```

```
CREATE TABLE Medical_History
(
  MedicalID Varchar(20) NOT NULL PRIMARY KEY,
  InsuranceID Varchar(20) NOT NULL REFERENCES Insurance(InsuranceID),
  Condition_Name Varchar(MAX)
);
```

```
ALTER TABLE Citizen
ADD GovernmentID VARBINARY(250);
```

Computed columns based on Functions - to calculate age

```
Create Function AgeCitizen(@DOB Date)
Returns INT
As
Begin
Declare @Age INT
Set @Age = (datediff(hour,@DOB,getdate())/8766)
Return @Age;
END
```

Implementation of Column level Encryption on Government ID

```
ALTER TABLE Citizen
ADD GovernmentID VARBINARY(250);

UPDATE Citizen
SET GovernmentID = EncryptByKey(Key_GUID(N'EncryptionKey'),CONVERT(VARBINARY,'G01'))
WHERE CitizenID=18;
UPDATE Citizen
SET GovernmentID = EncryptByKey(Key_GUID(N'EncryptionKey'),CONVERT(VARBINARY,'G02'))
WHERE CitizenID=19;
UPDATE Citizen
SET GovernmentID = EncryptByKey(Key_GUID(N'EncryptionKey'),CONVERT(VARBINARY,'G03'))
WHERE CitizenID=20;
```

Table level constraint based on function for vacant beds in hospital

```
create function checkHospitalbed(@HID varchar(5))
returns smallint
as
begin
    declare @count smallint=0;
    select @count=count(HospitalID)
    from Hospital
    where HospitalID=@HID
    and vacantBeds=0;
    return @count;
End;

Alter table Positive add constraint NoAdmission check (dbo.checkhospitalbed(HospitalID)=0);

insert into POSITIVE values('P1201','T40','H10','mild','active','2020-8-16')
```

{ OUTPUT }

Messages

Msg 547, Level 16, State 0, Line 693

The INSERT statement conflicted with the CHECK constraint "NoAdmission". The conflict occurred in database "CTrace", table "dbo.POSITIVE", column 'HospitalID'. The statement has been terminated.

Trigger for vacant units in Public Quarantine

```
CREATE TRIGGER trg_InsteadOf_Positive_PQ
ON Out_Patient
INSTEAD OF INSERT
AS
BEGIN
    SET NOCOUNT ON;
    Declare @QuarantineID varchar(10)=''
    Declare @VacantUnits int=0

    Select @QuarantineID = isnull(pq.QuarantineID,''),@VacantUnits=isnull(pq.VacantUnits,0)
    from Public_Quarantine pq
    join inserted I
    on I.QuarantineID = pq.QuarantineID

    if(@QuarantineID='')
    Begin
        Raiserror('Invalid Public Quarantine Name. Statement Terminated OK', 16, 1)
        return
    End
    if(@VacantUnits=0)
    Begin
        Raiserror('There is no bed in the Public Quarantine. Statement Terminated OK', 16, 1)
        return
    End
End
```

```
INSERT INTO Out_Patient(PatientID, Checkback_Date, QuarantineID)
Select PatientID ,GETDATE(),@QuarantineID
FROM inserted

if(@@ERROR=0) -- No Error
Begin
    update Public_Quarantine
    set VacantUnits=VacantUnits-1 where @QuarantineID=QuarantineID
End
END
```

{ OUTPUT }



```
Msg 50000, Level 16, State 1, Procedure trg_InsteadOf_Positive_PQ, Line 22 [Batch Start Line 36]
There is no bed in the Public Quarantine. Statement Terminated OK
```

```
(1 row affected)
```

```
Completion time: 2020-08-16T10:35:59.1388818-07:00
```


Report for Zones using views

```
CREATE VIEW [dbo].[vwPercentage]
AS
WITH Temp AS
(
SELECT a.City AS [City], a.Country AS [Country],
(SELECT COUNT(po.PatientID)
FROM POSITIVE po JOIN Tested te ON po.TestedID=te.TestedID
LEFT JOIN CitizenAddress ca ON ca.CitizenID = te.CitizenID
LEFT JOIN Address ad ON ad.AddressID = ca.AddressID
WHERE po.Patient_Status = 'Active' AND ad.City=a.City) AS Active_Count,
COUNT(C.CitizenID) AS Cit_count,
(
((SELECT COUNT(po.PatientID)
FROM POSITIVE po JOIN Tested te ON po.TestedID=te.TestedID
LEFT JOIN CitizenAddress ca ON ca.CitizenID = te.CitizenID
LEFT JOIN Address ad ON ad.AddressID = ca.AddressID
WHERE po.Patient_Status = 'Active' AND ad.City=a.City) * 100) / (COUNT(C.CitizenID))
) AS Percentage
FROM Positive P
LEFT JOIN Tested T
ON P.TestedID = T.TestedID
LEFT JOIN Citizen C
ON C.CitizenID = T.CitizenID
LEFT JOIN CitizenAddress ca
ON ca.CitizenID = T.CitizenID
LEFT JOIN Address a
ON a.AddressID = ca.AddressID
WHERE City IS NOT NULL
GROUP BY a.City, a.Country
)
SELECT Country, City, Cit_count, Active_Count, Percentage,
```

	Country	City	Cit_count	Active_Count	Percentage	Zone
1	US	Bellingham	5	2	40	CRITICAL
2	US	Bonney Lake	3	1	33	MODERATE
3	US	Lake Stevens	2	1	50	CRITICAL
4	US	Maple Valley	3	1	33	MODERATE
5	US	Renton	6	1	16	SAFE
6	US	Wenatchee	8	3	37	MODERATE

{ OUTPUT }

```
[Zone] =
CASE
WHEN Percentage < 30 THEN 'SAFE'
WHEN Percentage > 30 AND [Percentage] < 40 THEN 'MODERATE'
WHEN Percentage >= 40 THEN 'CRITICAL'
END
FROM Temp
GO
```

```

create view Home_Quarantine
as
with temp as
(
select p.patientid,c.FirstName,c.LastName,c.Gender,a.Housenumber,a.streetname,a.city
from positive p
join tested t
on t.TestedID= p.TestedID
join Citizen c
on c.CitizenID=t.CitizenID
join CitizenAddress ca
on ca.CitizenID=c.CitizenID
join Address a
on a.addressid=ca.addressid
where p.PatientID
Not IN(select PatientID from Out_Patient)
)
select * from temp where PatientID Not IN(select patientid from Hospitalized_Patient);

select * from Home_Quarantine

```

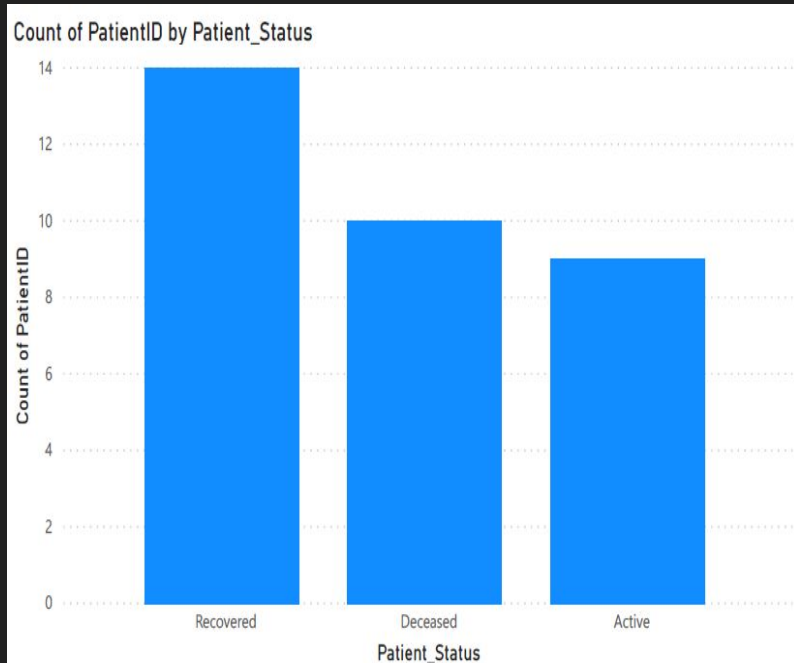
Report on Home Quarantine using views

{ OUTPUT } →

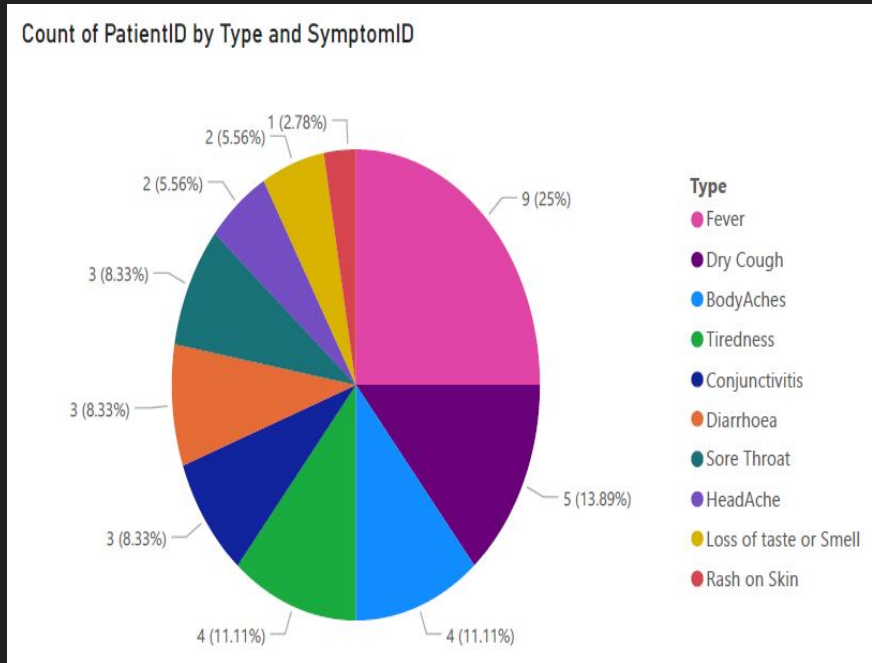
	patientid	FirstName	LastName	Gender	Housenumber	streetname	city
1	P1	Nathan	Ake	Male	114	12th PI SW	Renton
2	P28	Nathan	Ake	Male	114	12th PI SW	Renton
3	P5	Recee	James	Male	116	Corwin PI S	Bellingham
4	P6	Anokhi	Lodha	Female	118	Corson Avenue S	Bellingham
5	P7	Ivanka	Trump	Female	118	Corson Avenue S	Bellingham
6	P8	Kate	Winset	Female	118	Corson Avenue S	Bellingham
7	P9	Rahul	Johar	Male	120	Corporate Drive N	Bonney La

Reports

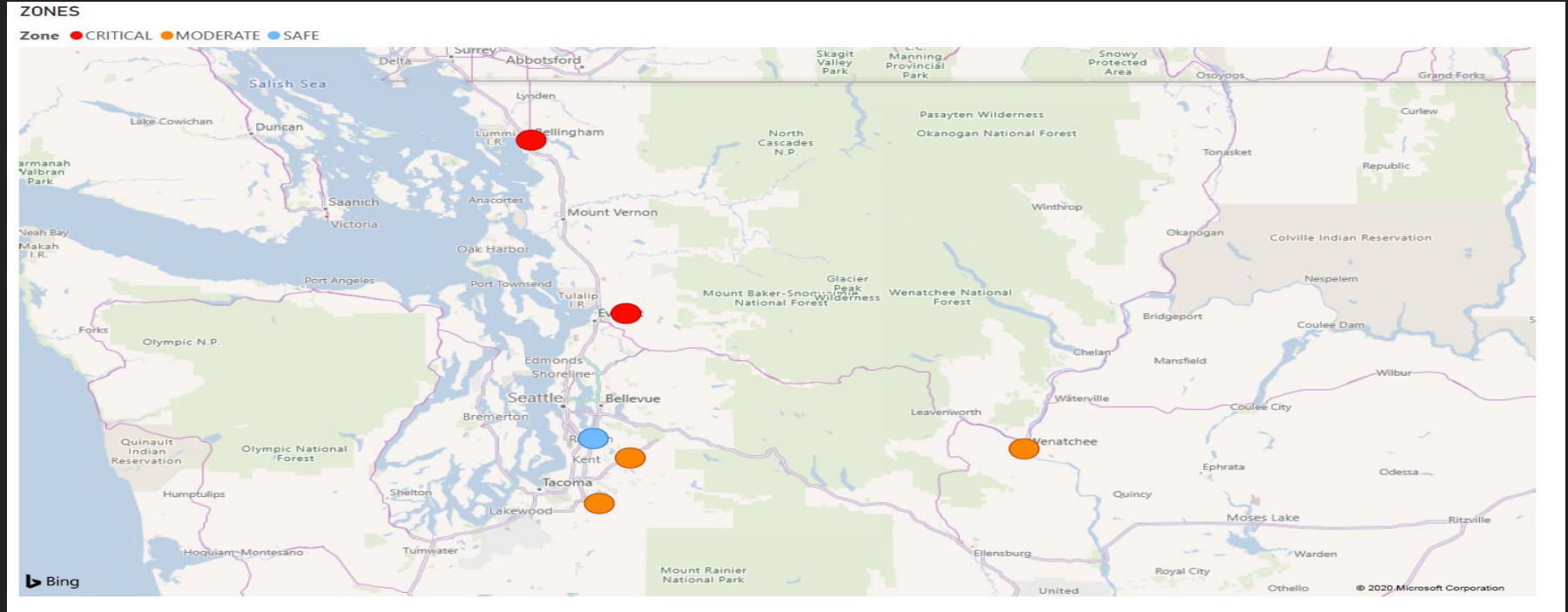
Column Chart - Representing status of patients



Pie Chart - Most common symptoms



Map representation of Critical to Safe Zones (Washington US)

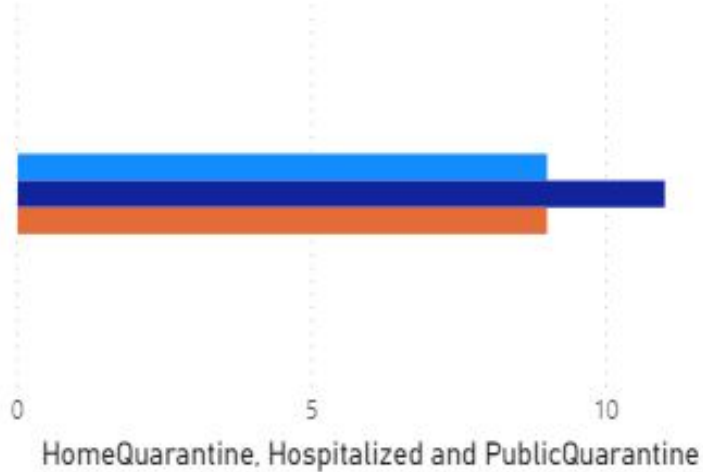


- Safe Zone : Total Percentage of positive patients below 20% of Total Population
- Moderate Zone : Total Percentage of positive patients above 20% and below 40% of Total Population
- Critical Zone : Total Percentage of positive patients above 40% of Total Population

Clustered Bar Chart - Representing the Quarantine centers

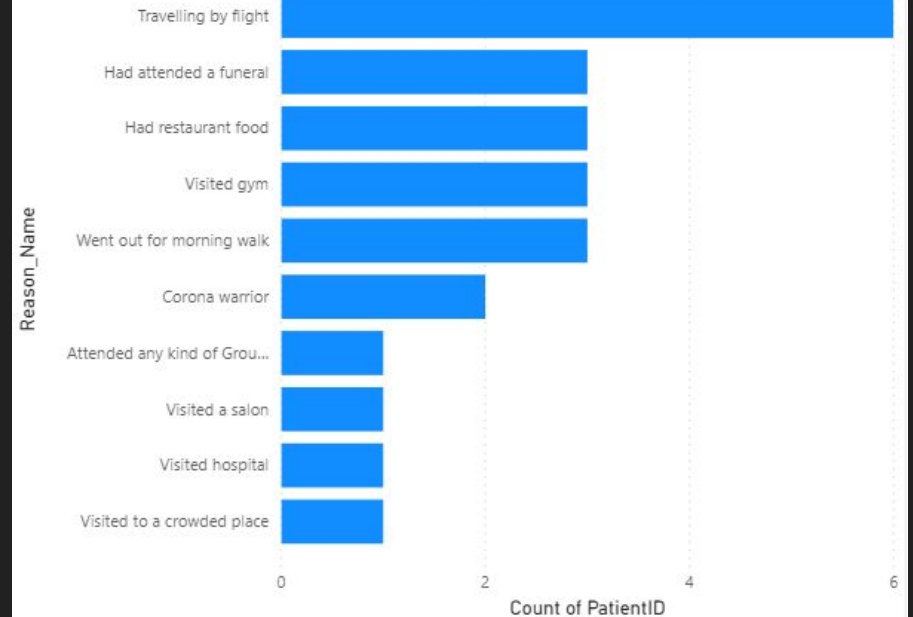
HomeQuarantine, Hospitalized and
PublicQuarantine

● HomeQuarantine ● Hospitalized ● PublicQuarantine



Bar Chart - Representing the most common reason for Covid

Count of PatientID by Reason_Name



CONCLUSION

- Contact Tracing as a whole not only help us to trace citizens exposed to a Covid-19 positive patient but also allow us to have following functionalities:
 - It also enable us to categorize areas in a state as critical, safe and moderate zones.
 - It also help us to have whole record about a patient like hospital he is admitted in, medical history, potential reason of infection etc.
- Present Scope of Contact Tracing is within Washington State, USA.
- The **future scope** of Contact Tracing is that we can extend its Implementation for each and every country in the world.