

DAMG 6210 DATA MGT AND DATABASE DESIGN

# Waste Management System

AMISHA GOKHALE  
VIDYAVATI PULLI  
KOUSHAL SAI AMSHALA  
DEEP SHAH





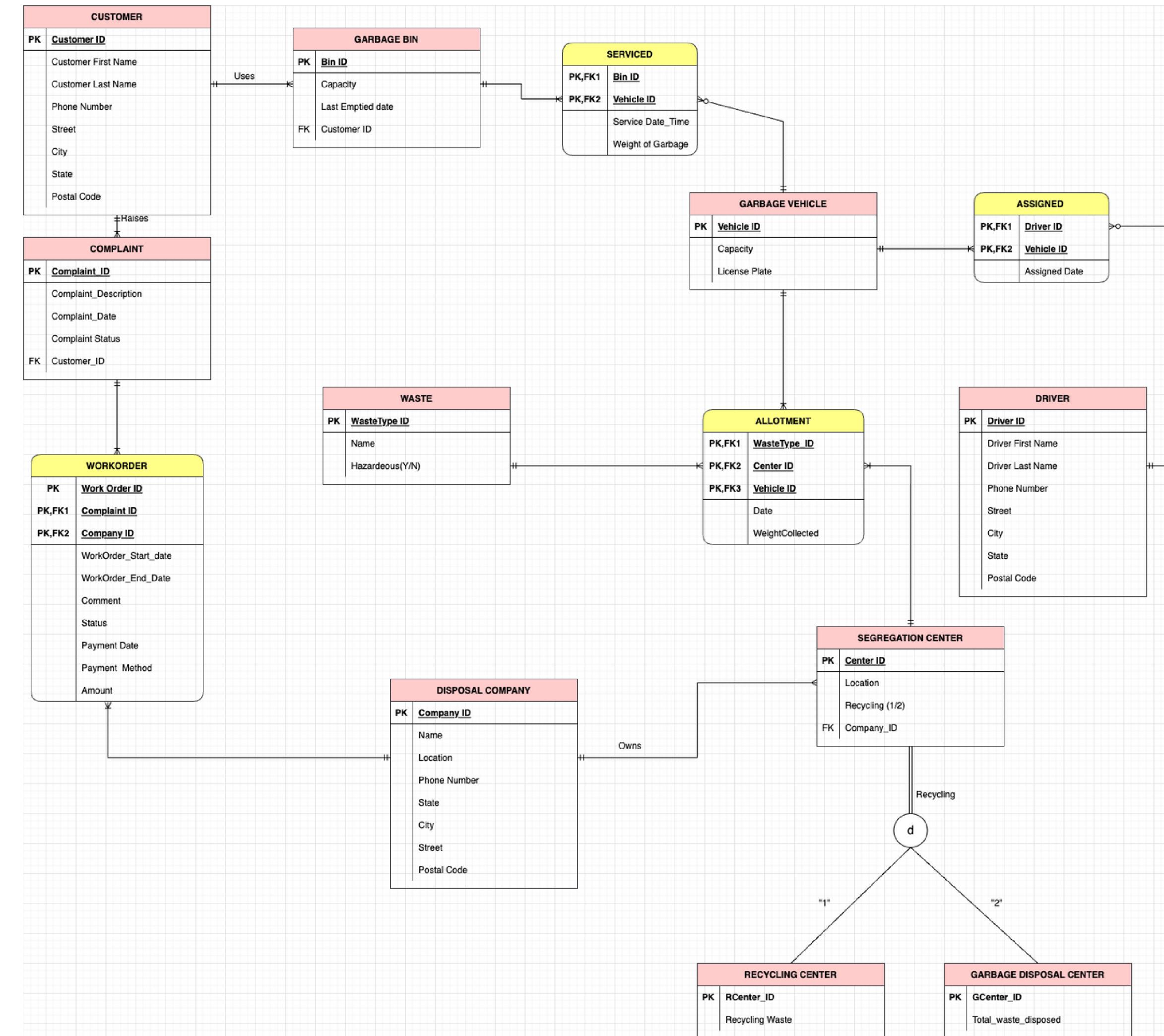
# OBJECTIVE

- Enables efficient storage, retrieval, and analysis of Waste types and quantities of waste collected
- This Database includes identifying patterns and trends, tracking waste disposal methods, and improving overall waste management practices.

# BUSINESS PROBLEM ADDRESS

- The current waste management system lacks a storing, retrieving and drawing insights about various types and weights of waste from collection to disposal.
- There is a need to develop innovative solutions that can optimize waste management processes, reduce waste generation, and promote the circular economy.

# ERD



# SAMPLE DDL

```
CREATE TABLE Customer (
    CustomerID INT NOT NULL IDENTITY(1, 1),
    CustomerFirstName NVARCHAR(25),
    CustomerLastName NVARCHAR(25),
    CustomerPhoneNumber CHAR(10),
    CustomerStreet VARCHAR(30),
    CustomerCity VARCHAR(20),
    CustomerState CHAR(2),
    CustomerPostalCode CHAR(5),
    CONSTRAINT Customer_PK PRIMARY KEY (CustomerID),
    CONSTRAINT chk_Custphone CHECK (CustomerPhoneNumber not like '%[^0-9]%' ),
    CONSTRAINT chk_zip CHECK (CustomerPostalCode not like '%[^0-9]%' )
);
```

```
INSERT INTO Customer (CustomerFirstName, CustomerLastName, CustomerPhoneNumber, CustomerStreet, CustomerCity, CustomerState, CustomerPostalCode)
VALUES ('John', 'Doe', '1234567890', 'Main St', 'New York City', 'NY', '10001');
INSERT INTO Customer (CustomerFirstName, CustomerLastName, CustomerPhoneNumber, CustomerStreet, CustomerCity, CustomerState, CustomerPostalCode)
VALUES ('Jane', 'Smith', '2345678901', 'Broadway', 'New York City', 'NY', '10002');
INSERT INTO Customer (CustomerFirstName, CustomerLastName, CustomerPhoneNumber, CustomerStreet, CustomerCity, CustomerState, CustomerPostalCode)
VALUES ('Mike', 'Johnson', '3456789012', '5th Ave', 'New York City', 'NY', '10003');
INSERT INTO Customer (CustomerFirstName, CustomerLastName, CustomerPhoneNumber, CustomerStreet, CustomerCity, CustomerState, CustomerPostalCode)
VALUES ('Emily', 'Wang', '4567890123', 'Madison Ave', 'New York City', 'NY', '10004');
INSERT INTO Customer (CustomerFirstName, CustomerLastName, CustomerPhoneNumber, CustomerStreet, CustomerCity, CustomerState, CustomerPostalCode)
VALUES ('Tom', 'Lee', '5678901234', 'Lexington Ave', 'New York City', 'NY', '10005');
INSERT INTO Customer (CustomerFirstName, CustomerLastName, CustomerPhoneNumber, CustomerStreet, CustomerCity, CustomerState, CustomerPostalCode)
VALUES ('David', 'Brown', '6789012345', 'Market St', 'San Francisco', 'CA', '94102');
INSERT INTO Customer (CustomerFirstName, CustomerLastName, CustomerPhoneNumber, CustomerStreet, CustomerCity, CustomerState, CustomerPostalCode)
VALUES ('Sarah', 'Wilson', '7890123456', 'Mission St', 'San Francisco', 'CA', '94103');
INSERT INTO Customer (CustomerFirstName, CustomerLastName, CustomerPhoneNumber, CustomerStreet, CustomerCity, CustomerState, CustomerPostalCode)
VALUES ('Jessica', 'Garcia', '8901234567', 'Geary Blvd', 'San Francisco', 'CA', '94104');
INSERT INTO Customer (CustomerFirstName, CustomerLastName, CustomerPhoneNumber, CustomerStreet, CustomerCity, CustomerState, CustomerPostalCode)
VALUES ('Eric', 'Taylor', '9012345678', 'Civic Center Dr', 'Beverly Hills', 'CA', '90210');
INSERT INTO Customer (CustomerFirstName, CustomerLastName, CustomerPhoneNumber, CustomerStreet, CustomerCity, CustomerState, CustomerPostalCode)
VALUES ('Ashley', 'Rodriguez', '0123456789', 'Wilshire Blvd', 'Beverly Hills', 'CA', '90211');
INSERT INTO Customer (CustomerFirstName, CustomerLastName, CustomerPhoneNumber, CustomerStreet, CustomerCity, CustomerState, CustomerPostalCode)
VALUES ('John', 'Doe', '1234567890', '123 Main St', 'Chicago', 'IL', '60601');
INSERT INTO Customer (CustomerFirstName, CustomerLastName, CustomerPhoneNumber, CustomerStreet, CustomerCity, CustomerState, CustomerPostalCode)
VALUES ('Jane', 'Smith', '2345678901', '456 State St', 'Springfield', 'IL', '62701');
INSERT INTO Customer (CustomerFirstName, CustomerLastName, CustomerPhoneNumber, CustomerStreet, CustomerCity, CustomerState, CustomerPostalCode)
VALUES ('Mike', 'Johnson', '3456789012', '789 Elm St', 'Peoria', 'IL', '61601');
INSERT INTO Customer (CustomerFirstName, CustomerLastName, CustomerPhoneNumber, CustomerStreet, CustomerCity, CustomerState, CustomerPostalCode)
VALUES ('Sarah', 'Brown', '4567890123', '321 Oak St', 'Rockford', 'IL', '61101');
INSERT INTO Customer (CustomerFirstName, CustomerLastName, CustomerPhoneNumber, CustomerStreet, CustomerCity, CustomerState, CustomerPostalCode)
VALUES ('David', 'Lee', '5678901234', '654 Pine St', 'Decatur', 'IL', '62521');
INSERT INTO Customer (CustomerFirstName, CustomerLastName, CustomerPhoneNumber, CustomerStreet, CustomerCity, CustomerState, CustomerPostalCode)
VALUES ('Emily', 'Jones', '5123456789', '1234 Main St', 'Austin', 'TX', '78701');
INSERT INTO Customer (CustomerFirstName, CustomerLastName, CustomerPhoneNumber, CustomerStreet, CustomerCity, CustomerState, CustomerPostalCode)
VALUES ('Michael', 'Garcia', '7134567890', '5678 Elm St', 'Houston', 'TX', '77002');
INSERT INTO Customer (CustomerFirstName, CustomerLastName, CustomerPhoneNumber, CustomerStreet, CustomerCity, CustomerState, CustomerPostalCode)
VALUES ('Rachel', 'Sanchez', '2145678901', '9101 Oak Blvd', 'Dallas', 'TX', '75201');
INSERT INTO Customer (CustomerFirstName, CustomerLastName, CustomerPhoneNumber, CustomerStreet, CustomerCity, CustomerState, CustomerPostalCode)
VALUES ('William', 'Hernandez', '8176789012', '3456 Cedar Ave', 'Fort Worth', 'TX', '76102');
INSERT INTO Customer (CustomerFirstName, CustomerLastName, CustomerPhoneNumber, CustomerStreet, CustomerCity, CustomerState, CustomerPostalCode)
VALUES ('Samantha', 'Martinez', '2107890123', '7890 Pine St', 'San Antonio', 'TX', '78201');
```

# PROCEDURES

```
-- Procedure
CREATE PROC updateDate
(
    @workId int,
    @status VARCHAR(10),
    @endDate DATE
)
AS
BEGIN
    UPDATE WorkOrder
    SET WorkOrderEndDate = @endDate
    WHERE WorkOrderID = @workId AND WorkOrderStatus = @status
END

EXEC updateDate @workId = 21, @status = 'Active', @endDate = '2024-03-09'
```

```
1  CREATE PROC totalWasteByWasteID (
2      @wasteID INT,
3      @TotalWasteGenerated INT OUTPUT
4  )
5  AS
6  BEGIN
7      SELECT @TotalWasteGenerated = SUM(a.WeightCollected)
8      From Allotment a
9      join Waste w
10     on w.WasteID = a.WasteID
11     WHERE a.wasteID = @wasteID
12     group by w.wasteID
13
14 END
15
16 DECLARE @TotalWasteGenerated INT
17 EXEC totalWasteByWasteID 3, @TotalWasteGenerated OUT
18 PRINT @TotalWasteGenerated
19
```

# TRIGGERS

```
CREATE TRIGGER CheckVehicleCapacity
ON Vehicle
AFTER INSERT, UPDATE
AS
BEGIN
    IF EXISTS (SELECT 1 FROM inserted WHERE VehicleCapacity > 10000)
    BEGIN
        RAISERROR('Vehicle capacity cannot be more than 30000', 16, 1)
        ROLLBACK TRANSACTION
    END
END
```

```
CREATE TRIGGER UpdateWorkOrderStatus
ON WorkOrder
AFTER UPDATE
AS
BEGIN
    IF UPDATE(WorkOrderEndDate)
    BEGIN
        UPDATE WorkOrder
        SET WorkOrderStatus = 'Completed'
        FROM inserted
        WHERE inserted.WorkOrderID = WorkOrder.WorkOrderID
              AND inserted.WorkOrderEndDate IS NOT NULL
    END
END
```

# COLOUUM BASED USER DEFINED FUNCTION

```
CREATE FUNCTION dbo.GetTotalWasteByCenterID (@CenterID INT)
RETURNS DECIMAL(10,2)
AS
BEGIN
    DECLARE @TotalWaste DECIMAL(10,2)
    SELECT @TotalWaste = SUM(GarbageWeight)
    FROM Serviced s
    JOIN Bin b ON s.BinID = b.BinID
    JOIN SegCenter c ON b.CustomerID = c.CompanyID
    WHERE c.CenterID = @CenterID
    RETURN @TotalWaste
END
```

```
ALTER TABLE SegCenter
ADD TotalWaste AS dbo.GetTotalWasteByCenterID(CenterID)

SELECT dbo.GetTotalWasteByCenterID(7) AS TotalWaste
```

## VIEWS

```
-- View of all drivers with their assigned vehicles:

CREATE VIEW DriverVehicles AS
SELECT d.DriverID, d.DriverFirstName, d.DriverLastName, d.DriverPhoneNumber, d.DriverStreet, d.DriverCity, d.DriverState, d.DriverPostalCode, a.VehicleID, a.AssignedDate
FROM Driver d
JOIN Assigned a ON d.DriverID = a.DriverID;

Select DriverID, DriverFirstName from DriverVehicles

-- View of all complaints and the corresponding customer information

CREATE VIEW Complaints AS
SELECT c.ComplaintID, c.ComplaintDescription, c.ComplaintDate, c.ComplaintStatus, c.CustomerID, cust.CustomerFirstName, cust.CustomerLastName, cust.CustomerPhoneNumber, cust.CustomerStreet, cust.CustomerCity, cust.CustomerState, cust.CustomerPostalCode
FROM Complaint c
JOIN Customer cust ON c.CustomerID = cust.CustomerID;

Select ComplaintID, ComplaintStatus from Complaints

-- View of all companies with their recycling centers

CREATE VIEW CompanyRecycling AS
SELECT comp.CompanyID, comp.CompanyName, comp.CompanyPhoneNumber, comp.CompanyState, comp.CompanyCity, comp.CompanyStreet, comp.CompanyPostalCode, rc.CenterID, rc.CenterLocation
FROM Company comp
JOIN SegCenter rc ON comp.CompanyID = rc.CompanyID;

select CompanyID, CompanyName, CompanyCity from CompanyRecycling

-- View of all recycling centers with the total waste recycled

CREATE VIEW RecyclingTotals AS
SELECT rc.CenterID, rc.RecyclingID, rc.TotalWasteRecycled
FROM RecyclingCenter rc;

select CenterID, TotalWasteRecycled from RecyclingTotals
```

# ENCRYPTION



## Waste Management

### Summary for Waste

5,400

Total Waste Disposed

4,900

Total Waste Recycled

10,300

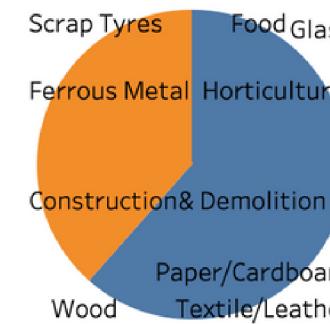
Total Waste

Total Waste collected across all centers.

Recycling ID

5,300.0

5,000.0



### Weight of Garbage collected by Waste Type



### Waste Collected at each center in each year

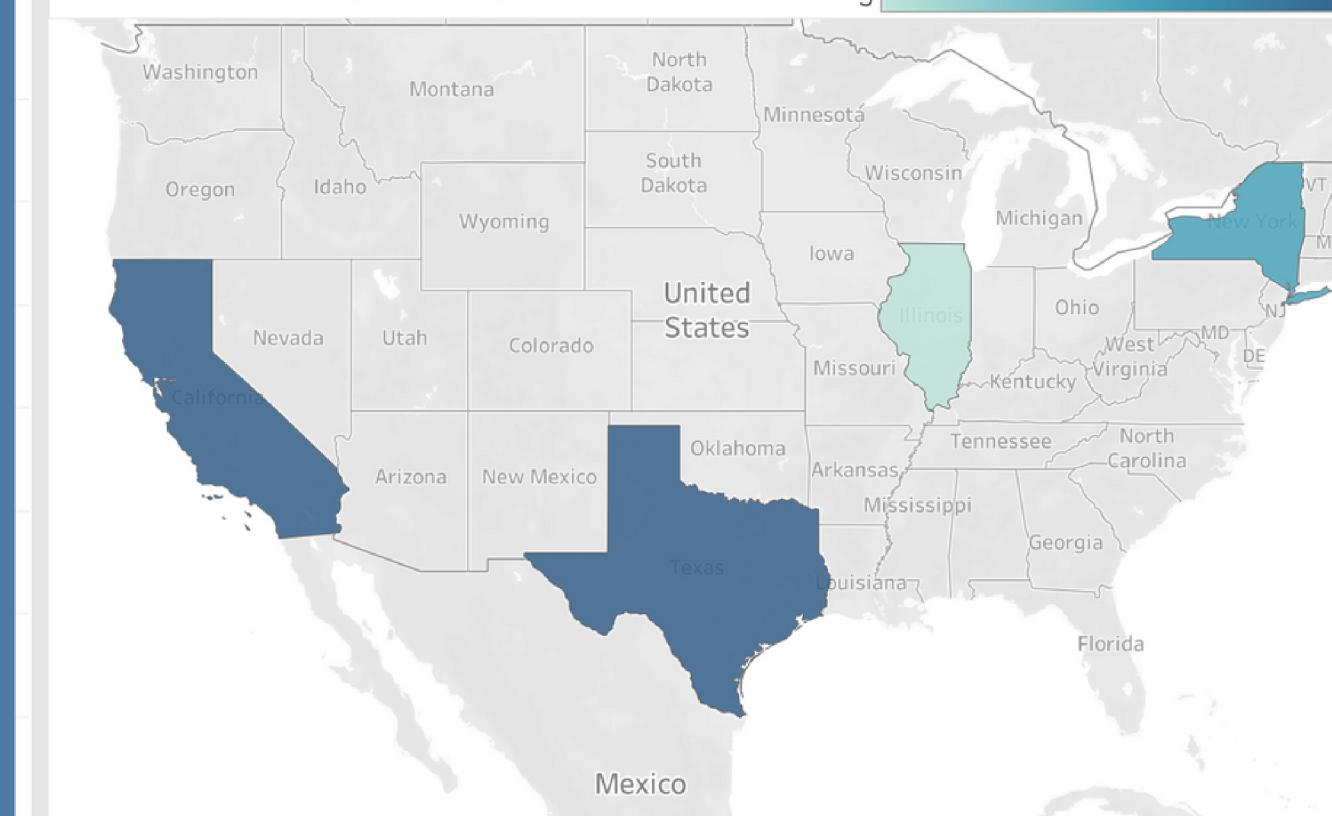
Year of Alloc.



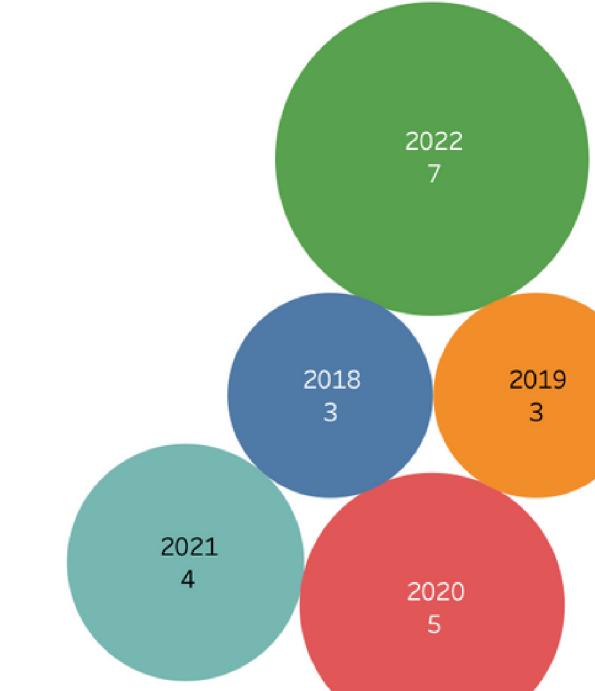
Total Waste

1500  
2000  
2500  
3000  
3500  
4000

### Number of Complaints per State



### Work Order Percent yearly





# WASTE MANAGEMENT



THANK YOU