Name : Haroon Ali Imran

Roll : 22024119-027

Section : BSCS-A

**Project Title : Town Management (Smart City)**

**Introduction**

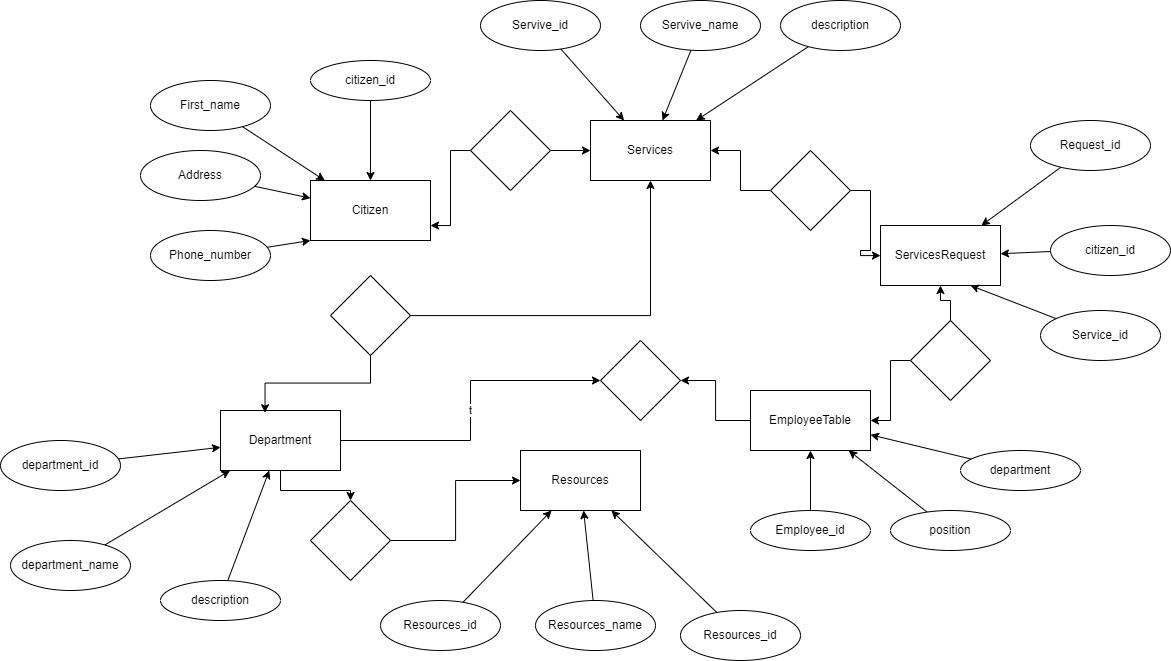
This database project involves the creation and management of a comprehensive database named Town\_Mangement for a fictional town. The database is designed to manage information about citizens, services provided, service requests, employees, departments, and resources. The main objective is to streamline town management by efficiently storing and retrieving relevant data. The database is built using SQL and includes several tables, procedures, and sample data to demonstrate its functionality.

**Database Schema**

1. **Database Creation and Usage**
   * The database Town\_Mangement is created and selected for use.
2. **Tables**
   * **Citizens**: Stores information about the citizens of the town.
   * **Services**: Contains details about various services provided by the town.
   * **Service\_Requests**: Records requests made by citizens for services.
   * **Employees**: Contains information about town employees.
   * **Departments**: Stores details of different departments in the town administration.
   * **Resources**: Manages the town's resources and their locations.
3. **Stored Procedures**
   * Procedures to insert data into the Citizens, Services, Service\_Requests, Employees, Departments, and Resources tables.
   * Procedures to retrieve all records from the Citizens, Services, and Service\_Requests tables.
4. **Sample Data**
   * Insert sample records into the Citizens, Services, Service\_Requests, Employees, Departments, and Resources tables.

**Entity-Relationship Diagram (ERD)**

The following ERD represents the schema of the Town\_Mangement database. It includes the relationships between the various tables such as Citizens, Services, Service\_Requests, Employees, Departments, and Resources.

****

**SQL Code**

create database Town\_Mangement;

use Town\_Mangement;

drop database Town\_Mangement;

CREATE TABLE Citizens (

citizen\_id INT PRIMARY KEY AUTO\_INCREMENT,

first\_name VARCHAR(50),

last\_name VARCHAR(50),

email VARCHAR(100),

phone VARCHAR(15),

address VARCHAR(200),

date\_of\_birth DATE

);

CREATE TABLE Services (

service\_id INT PRIMARY KEY AUTO\_INCREMENT,

service\_name VARCHAR(100),

description TEXT,

category VARCHAR(50)

);

CREATE TABLE Service\_Requests (

request\_id INT PRIMARY KEY AUTO\_INCREMENT,

citizen\_id INT,

service\_id INT ,

request\_date DATE,

status VARCHAR(50),

FOREIGN KEY (citizen\_id) REFERENCES Citizens(citizen\_id),

FOREIGN KEY (service\_id) REFERENCES Services(service\_id)

);

drop table Service\_Requests;

CREATE TABLE Employees (

employee\_id INT PRIMARY KEY AUTO\_INCREMENT,

first\_name VARCHAR(50),

last\_name VARCHAR(50),

position VARCHAR(50),

department VARCHAR(50)

);

CREATE TABLE Departments (

department\_id INT PRIMARY KEY AUTO\_INCREMENT,

department\_name VARCHAR(100),

description TEXT

);

CREATE TABLE Resources (

resource\_id INT PRIMARY KEY AUTO\_INCREMENT,

resource\_name VARCHAR(100),

quantity INT,

location VARCHAR(100)

);

DELIMITER //

CREATE PROCEDURE InsertCitizen (

IN first\_name VARCHAR(50),

IN last\_name VARCHAR(50),

IN email VARCHAR(100),

IN phone VARCHAR(15),

IN address VARCHAR(200),

IN date\_of\_birth DATE

)

BEGIN

INSERT INTO Citizens (first\_name, last\_name, email, phone, address, date\_of\_birth)

VALUES (first\_name, last\_name, email, phone, address, date\_of\_birth);

END //

DELIMITER ;

DELIMITER //

CREATE PROCEDURE InsertService (

IN service\_name VARCHAR(100),

IN description TEXT,

IN category VARCHAR(50)

)

BEGIN

INSERT INTO Services (service\_name, description, category)

VALUES (service\_name, description, category);

END //

DELIMITER ;

DELIMITER //

CREATE PROCEDURE InsertServiceRequest (

IN citizen\_id INT,

IN service\_id INT,

IN request\_date DATE,

IN status VARCHAR(50)

)

BEGIN

INSERT INTO Service\_Requests (citizen\_id, service\_id, request\_date, status)

VALUES (citizen\_id, service\_id, request\_date, status);

END //

DELIMITER ;

DELIMITER //

CREATE PROCEDURE InsertEmployee (

IN first\_name VARCHAR(50),

IN last\_name VARCHAR(50),

IN position VARCHAR(50),

IN department VARCHAR(50)

)

BEGIN

INSERT INTO Employees (first\_name, last\_name, position, department)

VALUES (first\_name, last\_name, position, department);

END //

DELIMITER ;

DELIMITER //

CREATE PROCEDURE InsertDepartment (

IN department\_name VARCHAR(100),

IN description TEXT

)

BEGIN

INSERT INTO Departments (department\_name, description)

VALUES (department\_name, description);

END //

DELIMITER ;

DELIMITER //

CREATE PROCEDURE InsertResource (

IN resource\_name VARCHAR(100),

IN quantity INT,

IN location VARCHAR(100)

)

BEGIN

INSERT INTO Resources (resource\_name, quantity, location)

VALUES (resource\_name, quantity, location);

END //

DELIMITER ;

DELIMITER //

CREATE PROCEDURE GetAllCitizens()

BEGIN

SELECT \* FROM Citizens;

END //

DELIMITER ;

DELIMITER //

CREATE PROCEDURE GetAllServices()

BEGIN

SELECT \* FROM Services;

END //

DELIMITER ;

DELIMITER //

CREATE PROCEDURE GetAllServiceRequests()

BEGIN

SELECT \* FROM Service\_Requests;

END //

DELIMITER ;

-- Insert sample citizens

CALL InsertCitizen('Ali', 'Hamza', 'AliHamza@gmail.com', '+1234567890', '123 Elm Street', '1980-01-01');

CALL InsertCitizen('Ali', 'Haider', 'Ali@gmail.com', '+1234567891', '456 Oak Avenue', '1990-02-02');

-- Insert sample services

CALL InsertService('Garbage Collection', 'Weekly garbage collection service', 'Sanitation');

CALL InsertService('Water Supply', 'Daily water supply service', 'Utilities');

-- Insert sample service requests

CALL InsertServiceRequest(1, 1, '2024-06-01', 'Pending');

CALL InsertServiceRequest(2, 2, '2024-06-02', 'Completed');

-- Insert sample employees

CALL InsertEmployee('Ali', 'Brown', 'Manager', 'Sanitation');

CALL InsertEmployee('Haroon', 'White', 'Technician', 'Utilities');

-- Insert sample departments

CALL InsertDepartment('Sanitation', 'Handles waste management and recycling services');

CALL InsertDepartment('Utilities', 'Manages water, electricity, and other utilities');

-- Insert sample resources

CALL InsertResource('Truck', 5, 'Garage 1');

CALL InsertResource('Water Pump', 10, 'Storage Facility');

-- Verify citizens

SELECT \* FROM Citizens;

-- Verify services

SELECT \* FROM Services;

-- Verify service requests

SELECT \* FROM Service\_Requests;

-- Verify employees

SELECT \* FROM Employees;

-- Verify departments

SELECT \* FROM Departments;

-- Verify resources

SELECT \* FROM Resources;