# DATABASE MANAGEMENT SYSTEM - CSA0593 ASSIGNMENT 3 R. YASHWANTH VARMA 192311392

## **QUESTION:**

Build a database to support a crowdsourced mapping platform for collecting and updating map data.

- Design tables for locations, users, map edits, and changesets.
- Write queries to identify the most recent map changes and verify data accuracy.
- Implement triggers to validate user edits and prevent duplicate entries.
- Discuss indexing strategies to enable fast retrieval of data for real-time map rendering

#### **ANSWER:**

# CONCEPTUAL ER DIAGRAM:

```
LOCATION
| LocationID (PK) |
Name
Coordinates
Type
                 1
Description
        1
        |----- MAP_EDIT
                        | EditID (PK)
                        | LocationID (FK) |
                        | UserID (FK)
                        | EditType
                        Timestamp
| UserID (PK)
Username
Email
| JoinDate
Reputation
        ----- CHANGESET
                        | ChangesetID (PK)|
                        | UserID (FK)
                        Description
                        Timestamp
```

### LOGICAL ER DIAGRAM:

```
LOCATION
| LocationID (PK) |----< MAP_EDIT
Name
Coordinates
                | EditID (PK)
                        | LocationID (FK) |
Type
Description
                        UserID (FK)
                        | EditType
                        Timestamp
USER
UserID (PK)
                ----< MAP_EDIT
Username
                ----< CHANGESET
Email
JoinDate
                        | ChangesetID (PK)|
Reputation
                        | UserID (FK)
                        Description
                        Timestamp
```

#### PHYSICAL ER DIAGRAM:

```
LOCATION
| LocationID (PK)
Name
                   VARCHAR(100) NOT NULL
Coordinates
                   GEOMETRY | -- For storing lat/lon
                   VARCHAR(50)
Type
Description
                   TEXT
           ----< MAP_EDIT
                         | EditID (PK)
                                                      1
                         | LocationID (FK)
                                                      1
                         UserID (FK)
                         | EditType
                                             VARCHAR(50)
                         Timestamp
                                             DATETIME |
UserID (PK)
Username
                   VARCHAR(50) UNIQUE NOT NULL
Email
                   VARCHAR (150) UNIQUE NOT NULL
JoinDate
                   DATETIME
Reputation
        1
            ----< CHANGESET
                         | ChangesetID (PK)
                         UserID (FK)
                         Description
                                             TEXT
                         Timestamp
                                             DATETIME |
```

# **SQL STATEMENTS:**

```
mysql
CREATE DATABASE MapData;
USE MapData;
CREATE TABLE Locations (
LocationID INT AUTO INCREMENT PRIMARY KEY,
Latitude DECIMAL(10, 8),
Longitude DECIMAL(11, 8),
LocationName VARCHAR(100)
);
CREATE TABLE Users (
UserID INT AUTO INCREMENT PRIMARY KEY,
Username VARCHAR(50),
Email VARCHAR(100)
);
CREATE TABLE MapEdits (
EditID INT AUTO INCREMENT PRIMARY KEY,
LocationID INT,
UserID INT,
EditDate DATETIME,
EditType VARCHAR(50),
FOREIGN KEY (LocationID) REFERENCES Locations(LocationID),
```

```
FOREIGN KEY (UserID) REFERENCES Users(UserID)
);
CREATE TABLE Changesets (
 ChangesetID INT AUTO INCREMENT PRIMARY KEY,
 EditID INT,
 ChangesetDate DATETIME,
 ChangesetDescription VARCHAR(255),
FOREIGN KEY (EditID) REFERENCES MapEdits(EditID)
);
Queries:
mysql
-- Most Recent Map Changes
SELECT
Locations.LocationName,
MapEdits.EditDate,
 MapEdits.EditType
FROM
 Locations
JOIN MapEdits ON Locations.LocationID = MapEdits.LocationID
ORDER BY
 MapEdits.EditDate DESC;
-- Verify Data Accuracy
SELECT
 Locations.LocationName,
```

```
MapEdits.EditType,
 Changesets.ChangesetDescription
FROM
 Locations
 JOIN MapEdits ON Locations.LocationID = MapEdits.LocationID
 JOIN Changesets ON MapEdits.EditID = Changesets.EditID
WHERE
 MapEdits.EditType = 'Update';
Triggers:
Mysql
DELIMITER //
CREATE TRIGGER tr_ValidateUserEdits
BEFORE INSERT ON MapEdits
FOR EACH ROW
BEGIN
 IF NEW.EditType NOT IN ('Create', 'Update', 'Delete') THEN
  SIGNAL SQLSTATE '45000' SET MESSAGE TEXT = 'Invalid edit type';
 END IF;
END //
CREATE TRIGGER tr PreventDuplicateEntries
BEFORE INSERT ON Locations
```

#### FOR EACH ROW

#### **BEGIN**

IF EXISTS (SELECT 1 FROM Locations WHERE Latitude = NEW.Latitude AND Longitude = NEW.Longitude) THEN

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Duplicate location entry';

END IF;

END //

DELIMITER;

## **Conclusion:**

This database design supports a crowdsourced mapping platform by storing locations, user information, map edits, and changesets. The queries enable identification of recent map changes and verification of data accuracy. The triggers validate user edits and prevent duplicate entries. Indexing strategies optimize data retrieval for real-time map rendering.