Trigger code:

```
DELIMITER //
CREATE TRIGGER UpdateDiseaseSeverity
AFTER INSERT ON UserConsultation
FOR EACH ROW
BEGIN
    DECLARE consultation_count INT;

    SELECT COUNT(*) INTO consultation_count
    FROM UserConsultation
    WHERE DiseaseID = NEW.DiseaseID;

    IF consultation_count > 5 THEN
        UPDATE Disease
        SET Severity = 'Severe'
        WHERE DiseaseID = NEW.DiseaseID;
    ELSEIF consultation_count > 3 THEN
        UPDATE Disease
        SET Severity = 'Moderate'
        WHERE DiseaseID = NEW.DiseaseID;
    ELSE
        UPDATE Disease
        SET Severity = 'Mild'
        WHERE DiseaseID = NEW.DiseaseID;
    END IF;
END;//
DELIMITER ;
```

Transaction code:

```
DELIMITER //
CREATE PROCEDURE DiseaseDiagnosis6(IN input_symptoms VARCHAR(1000), IN
max_results INT)
BEGIN
            SET TRANSACTION ISOLATION LEVEL READ COMMITTED;

    START TRANSACTION;

            DROP TEMPORARY TABLE IF EXISTS primary_diagnosis;
            DROP TEMPORARY TABLE IF EXISTS similar_diseases;
    -- first query + temp table
    CREATE TEMPORARY TABLE primary_diagnosis AS
    SELECT d.DiseaseID, d.DiseaseName, d.Description, COUNT(ds.SymptomID) as MatchingSymptomCount
    FROM Disease d
    JOIN Disease_Symptom ds ON d.DiseaseID = ds.DiseaseID
    WHERE ds.SymptomID IN (
       SELECT SymptomID FROM Symptom
       WHERE FIND_IN_SET(SymptomName, input_symptoms) > 0
    )
    GROUP BY d.DiseaseID, d.DiseaseName, d.Description
    ORDER BY MatchingSymptomCount DESC
    LIMIT max_results;

    -- second query + temp table
    CREATE TEMPORARY TABLE similar_diseases AS
    SELECT d1.DiseaseName as PrimaryDisease,
          d2.DiseaseName as SimilarDisease
    FROM Disease d1
    JOIN Disease_Symptom ds1 ON d1.DiseaseID = ds1.DiseaseID
    JOIN Disease_Symptom ds2 ON ds1.SymptomID = ds2.SymptomID
    JOIN Disease d2 ON ds2.DiseaseID = d2.DiseaseID
    WHERE d1.DiseaseName IN (SELECT DiseaseName FROM primary_diagnosis)
    AND d1.DiseaseID != d2.DiseaseID
    GROUP BY d1.DiseaseName, d2.DiseaseName;

    -- results of both query 1 and 2
    SELECT
       pd.DiseaseID,
       pd.DiseaseName,
       pd.Description,
       pd.MatchingSymptomCount,
       (SELECT GROUP_CONCAT(SimilarDisease SEPARATOR ', ')
        FROM similar_diseases
        WHERE PrimaryDisease = pd.DiseaseName) AS SimilarDiseases
    FROM primary_diagnosis pd;

    DROP TEMPORARY TABLE primary_diagnosis;
    DROP TEMPORARY TABLE similar_diseases;
    COMMIT;
END //
DELIMITER ;
```

Stored Procedure code:

```
USE sympchat_database;

DELIMITER //
CREATE PROCEDURE UserHealthHistory2(IN userid INT, IN monthsback INT)
BEGIN
    DECLARE user_exists INT DEFAULT 0;

    SELECT COUNT(*) INTO user_exists FROM User WHERE UserID = userid;
    IF user_exists = 0 THEN
        SELECT 'user not found' AS Error;
    ELSE
        IF monthsback IS NULL OR monthsback <= 0 THEN
            SET monthsback = 6;
        END IF;

                        -- first query
                        SELECT u.Username, s.SymptomName, COUNT(*) as SymptomFrequency
                        FROM User u
                        JOIN UserConsultation uc ON u.UserID = uc.UserID
                        JOIN UserConsultation_Symptom ucs ON uc.ConsultationID =
ucs.ConsultationID
                        JOIN Symptom s ON ucs.SymptomID = s.SymptomID
                        WHERE u.UserID = userid
                          AND uc.Date > (SELECT MAX(Date) - INTERVAL monthsback MONTH
FROM UserConsultation WHERE UserID = userid)
                        GROUP BY u.Username, s.SymptomName
                        HAVING COUNT(*) >= 1
                        ORDER BY SymptomFrequency DESC;

                        -- second query
                        SELECT
                                    u.Username,
                                    d.DiseaseName,
                                    COUNT(*) as DiagnosisFrequency,
                                    MIN(uc.Date) as FirstDiagnosis,
            MAX(uc.Date) as MostRecentDiagnosis,
            (
                SELECT GROUP_CONCAT(DISTINCT s.SymptomName
ORDER BY s.SymptomName SEPARATOR ', ')
                FROM UserConsultation_Symptom ucs
                JOIN Symptom s ON ucs.SymptomID = s.SymptomID
                WHERE ucs.ConsultationID IN (
                    SELECT ConsultationID
                    FROM UserConsultation uc2
                    WHERE uc2.UserID = userid
                    AND uc2.DiseaseID = uc.DiseaseID
                )
            ) as CommonSymptoms
        FROM User u
        JOIN UserConsultation uc ON u.UserID = uc.UserID
        JOIN Disease d ON uc.DiseaseID = d.DiseaseID
        WHERE u.UserID = userid
          AND uc.Date > (
              SELECT MAX(Date) - INTERVAL monthsback MONTH
              FROM UserConsultation
              WHERE UserID = userid
          )
        GROUP BY u.Username, d.DiseaseName, uc.DiseaseID
        HAVING COUNT(*) >= 1
        ORDER BY DiagnosisFrequency DESC, MostRecentDiagnosis DESC;
    END IF;
END //
DELIMITER ;
```