

TRƯỜNG ĐẠI HỌC KỸ THUẬT CÔNG NGHIỆP

KHOA ĐIỆN TỬ

BỘ MÔN: CÔNG NGHỆ THÔNG TIN



BÁO CÁO BÀI TẬP LỚN HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU

ĐỀ TÀI: QUẢN LÝ TÀI KHOẢN GAME

Giảng viên hướng dẫn	: ThS. ĐỖ DUY CÔP
Họ tên sinh viên	: TRẦN CHIẾN THẮNG
MSSV	: K215480106121
Lớp	: K57KMT.01

Thái Nguyên 2024

PHIẾU GIAO ĐỀ TÀI MÔN HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU

Sinh viên: Trần Chiến Thắng

MSSV: K215480106121

Lớp: K57KMT

Ngành: Kỹ thuật máy tính

Giáo viên hướng dẫn: ThS. Đỗ Duy Cốp

1. Tên đề tài: **“Quản lý tài khoản game”**.

2. Yêu cầu của phần mềm, ứng dụng:

- Tạo cơ sở dữ liệu cho bài toán gồm các bảng và thuộc tính bằng việc phân tích

3. Các sản phẩm, kết quả:

- Bản báo cáo, và cơ sở dữ liệu.

GIÁO VIÊN HƯỚNG DẪN

(Ký và ghi rõ họ tên)

ThS. Đỗ Duy Cốp

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

.....

.....

.....

.....

.....

.....

.....

.....

.....

Thái Nguyên, ngày...tháng.....năm.....

GIÁO VIÊN HƯỚNG DẪN

(Ký ghi rõ họ tên)

Github cá nhân

- Link: https://github.com/thangtran1904/QL_Tai_Khoan_Game/blob/main/README.md



- QR:

Thông tin cá nhân

- Họ và Tên: Trần Chiến Thắng
- MSSV: K215480106121
- Lớp: K57KMT
- Môn học: Hệ quản trị cơ sở dữ liệu

Tên bài toán

- Đề tài: Quản lý tài khoản game

Các chức năng

Bài toán quản lý tài khoản game ,thông tin nhân vật và quản lý lịch sử hoạt động

- Quản lý tài khoản : Gồm các thuộc tính tên đăng nhập,mật khẩu,ngày tạo tài khoản,lần đăng nhập cuối cùng.
- Có các chức năng thêm, sửa, xóa ,liệt kê theo tên đăng nhập,Đổi mật khẩu 1 tài khoản,check login cho 1 tài khoản.
- Quản lý thông tin nhân vật : Có các chức năng thêm, sửa, xóa, liệt kê theo profile_id .

- Quản lý lịch sử hoạt động : Có các chức năng thêm, sửa, xóa, liệt kê theo user_id.

Báo cáo

- Báo cáo số người dùng được tạo trong tháng
- Báo cáo danh sách các hoạt động của tài khoản trong tháng
- Báo cáo số lượng người dùng đã được tạo trong tháng hiện tại
- Báo cáo danh sách các người dùng được tạo trong tháng hiện tại
- Báo cáo số lượng người dùng được tạo trong từng ngày của tháng hiện tại

Các bảng của hệ thống

- nguoidung(#user_id,@Tendangnhap,@Matkhau,@Email,@NgaytaoTK,@Landangnhapcuoi,@TrangthaiTK)

	user_id	Tendangnhap	Matkhau	Email	NgaytaoTK	Landangnhapcuoi	TrangthaiTK
1	1	updaHoak	123456aa	th@gamil.com	2024-06-18 20:20:00.000	2024-06-19 10:01:50.000	Offline
2	2	updaHoak1	123456aA	tha@gmail.com	2024-06-14 10:25:40.000	2024-06-20 14:35:31.000	Offline
3	3	updaHoak3	123456Aa	than@gmail.com	2024-05-15 11:05:00.000	2024-06-19 10:10:00.000	Offline

The screenshot shows the SQL Server Enterprise Manager interface. The 'Object Explorer' on the left shows the database structure. The 'Table Designer' window is open for the 'dbo.nguoidung' table. The table has the following columns:

Column Name	Data Type	Allow Nulls
user_id	int	<input type="checkbox"/>
Tendangnhap	nvarchar(100)	<input type="checkbox"/>
Matkhau	nvarchar(255)	<input type="checkbox"/>
Email	nvarchar(100)	<input type="checkbox"/>
NgaytaoTK	datetime	<input type="checkbox"/>
Landangnhapcuoi	datetime	<input type="checkbox"/>
TrangthaiTK	nvarchar(50)	<input type="checkbox"/>

The 'Properties' window on the right shows the table's identity and other properties:

- Identity:** nguoidung
- Database Name:** quanlytalkhoangame
- Description:** quanlytalkhoangame
- Schema:** dbo
- Server Name:** quetaosocson/sqlpres
- Table Designer:**
 - Identity Column:** user_id
 - Indenable:** Yes
 - Lock Escalation:** Table
 - Regular Data Spac:** PRIMARY
 - Replicated:** No
 - Row GUID Column:**
 - Text/Image Filegr:** PRIMARY

- Các trường:
 - user_id (INT): Primary Key (PK). Đây là một số duy nhất cho mỗi người dùng.
 - Tendangnhap (NVARCHAR(100)): Candidate Key. Là tên đăng nhập duy nhất.

- Matkhau (NVARCHAR(255)): Mật khẩu được mã hóa (ví dụ: SHA2_256).
 - Email (NVARCHAR(100)): Địa chỉ email của người dùng.
 - NgaytaoTk (DATE): Ngày tạo tài khoản, có thể NULL nếu không có thông tin.
 - Landangnhapcuoi (DATETIME): Thời điểm đăng nhập cuối cùng.
 - Capnhatluc (DATETIME): Thời điểm cập nhật thông tin người dùng.
- Ràng buộc dữ liệu:
 - user_id: PK không được NULL, và giá trị duy nhất.
 - Tendangnhap: Candidate Key, không được NULL và duy nhất.
 - Matkhau: Bắt buộc nhập.
 - Email: Có thể NULL.
 - NgaytaoTk, Landangnhapcuoi, Capnhatluc: Có thể NULL tùy thuộc vào nhu cầu.
- Giải thích:
 - user_id là PK để xác định duy nhất mỗi người dùng.
 - Tendangnhap là Candidate Key để xác định duy nhất mỗi tên đăng nhập.
 - Matkhau được yêu cầu bắt buộc nhập để đảm bảo tính bảo mật.
 - Email có thể NULL vì người dùng có thể không cung cấp địa chỉ email.
 - NgaytaoTk, Landangnhapcuoi, Capnhatluc là các trường thời gian có thể NULL nếu không có thông tin hoặc chưa cập nhật.
- lichsuhoatdong(#log_id,@user_id,@Loaihoatdong,@ChitietHD,@Thoigian)

	log_id	user_id	Loaihoatdong	ChitietHD	Thoigian
1	1	123	Mua vàng	Đổi 20000000 VNĐ ra 100000 vàng	2024-06-18 21:20:00.000
2	2	124	Mua vàng	Đổi 5000000 VNĐ ra 25000 vàng	2024-06-19 17:00:50.000
3	3	125	Mua vàng	Đổi 1000000 VNĐ ra 5000 vàng	2024-06-19 10:20:00.000
4	4	123	Mua Hp 9x	Đổi 1000 vàng ra 10000 Hp 9x	2024-06-23 20:10:00.000
5	5	124	Mua thức ăn 9x	Đổi 100000 xu ra 9 thức ăn 9x	2024-06-25 10:02:00.000
6	6	124	Đổi xu	Đổi 1000 vàng ra 100000000 xu	2024-06-27 00:00:00.000

- Các trường:
 - log_id (INT): Primary Key (PK). Đây là một số duy nhất cho mỗi log hoạt động.
 - user_id (INT): Foreign Key (FK) tham chiếu đến nguoidung(user_id). Xác định người dùng thực hiện hoạt động.
 - Loaihoatdong (NVARCHAR(50)): Loại hoạt động được thực hiện.
 - ChitietHD (NVARCHAR(250)): Chi tiết về hoạt động.
 - Thoigian (DATETIME): Thời điểm hoạt động được thực hiện.
- Ràng buộc dữ liệu:
 - log_id: PK không được NULL và duy nhất.
 - user_id: FK để liên kết với bảng nguoidung(user_id).
 - Loaihoatdong, ChitietHD, Thoigian: Các trường này có thể NULL hoặc không tùy thuộc vào nhu cầu cụ thể của ứng dụng.
- Giải thích:

- log_id là PK để xác định duy nhất mỗi log hoạt động.
- user_id là FK để liên kết log hoạt động với người dùng tương ứng.
- Loaihoatdong và ChitietHD là các trường lưu trữ văn bản, còn Thoigian là trường thời gian để lưu lại thời điểm hoạt động được thực hiện.
- thongtinnhanvat(#profile_id,@user_id,@TenNV,@Capdo,@EXP,@Vip,@Danhchieu,@Tiendanap,@Tiendadung,@Capnhatluc)

	profile_id	user_id	TenNV	Capdo	EXP	Vip	Danhchieu	Tiendanap	Tiendadung	Capnhatluc
1	1	123	upda	90	51,06	6	Đại Gia	20000000	15000000	2024-06-18 10:10:00.000
2	2	124	upda1	89	60,2	4	Đại Gia Mới Nhũ	5000000	5000000	2024-06-19 16:50:50.000
3	3	125	upda2	87	88,62	2	Fan Cứng	1000000	1000000	2024-06-19 16:51:00.000

- Các trường:
 - profile_id (INT): Primary Key (PK). Đây là một số duy nhất cho mỗi thông tin nhân vật.
 - user_id (INT): Foreign Key (FK) tham chiếu đến nguoidung(user_id). Xác định người dùng liên quan đến thông tin nhân vật.
 - TenNV (NVARCHAR(100)): Tên của nhân vật.
 - Capdo (INT): Cấp độ của nhân vật.
 - EXP (FLOAT): Kinh nghiệm tích lũy của nhân vật.
 - Vip (INT): Mức độ VIP của nhân vật.
 - Danhchieu (NVARCHAR(50)): Danh hiệu của nhân vật.
 - Tiendanap (NVARCHAR(50)): Số tiền đã nạp vào tài khoản của nhân vật.

- Tiendadung (NVARCHAR(50)): Số tiền đã sử dụng của nhân vật.
- Capnhatluc (DATETIME): Thời điểm cập nhật thông tin nhân vật.
- Ràng buộc dữ liệu:
 - profile_id: PK không được NULL và duy nhất.
 - user_id: FK để liên kết với bảng nguoidung(user_id).
 - TenNV, Capdo, EXP, Vip, Danhhiu, Tiendanap, Tiendadung: Các trường này có thể NULL hoặc không tùy thuộc vào nhu cầu cụ thể của ứng dụng.
- Giải thích:
 - profile_id là PK để xác định duy nhất mỗi thông tin nhân vật.
 - user_id là FK để liên kết thông tin nhân vật với người dùng tương ứng.
 - Các trường như Capdo (INT), EXP (FLOAT), Capnhatluc (DATETIME) được sử dụng cho các giá trị số học, tiền tệ và thời gian, phù hợp với nhu cầu lưu trữ và tính toán trong ứng dụng.

Các Chức Năng

- Chức năng thêm các thuộc tính theo tên đăng nhập

```
CREATE PROCEDURE AddUserCharacterAndLog
    @Tendangnhap NVARCHAR(100),
    @Matkhan NVARCHAR(255),
    @Email NVARCHAR(100),
    @TenNV NVARCHAR(100),
    @Capdo INT,
    @EXP FLOAT,
    @Vip INT,
    @Danhhiu NVARCHAR(50),
    @Tiendanap NVARCHAR(50),
    @Tiendadung NVARCHAR(50),
    @Loaihoatdong NVARCHAR(50),
    @ChitietHD NVARCHAR(250)
AS
BEGIN
    SET NOCOUNT ON;

    BEGIN TRY
        BEGIN TRANSACTION;

        -- Step 1: Thêm người dùng
        DECLARE @PasswordHash NVARCHAR(255);
        SET @PasswordHash = HASHBYTES(SHA2_256, @Matkhan);

        INSERT INTO nguoidung (Tendangnhap, Matkhan, Email)
        VALUES (@Tendangnhap, @PasswordHash, @Email);

        DECLARE @user_id INT;
        SET @user_id = SCOPE_IDENTITY(); -- Lấy user_id được chèn lần cuối

        -- Step 2: Thêm hồ sơ nhân vật
        INSERT INTO thongtinnhanvat (user_id, TenNV, Capdo, EXP, Vip, Danhhiu, Tiendanap, Tiendadung)
        VALUES (@user_id, @TenNV, @Capdo, @EXP, @Vip, @Danhhiu, @Tiendanap, @Tiendadung);

        -- Step 3: Thêm nhật ký hoạt động
        INSERT INTO lichlichhoatdong (user_id, Loaihoatdong, ChitietHD, Thoigian)
        VALUES (@user_id, @Loaihoatdong, @ChitietHD, GETDATE());

        COMMIT TRANSACTION;
    END TRY
    BEGIN CATCH
        IF @@TRANCOUNT > 0
            ROLLBACK TRANSACTION;
        THROW;
    END CATCH;
END;
```

- Chức năng xóa các thuộc tính theo tên đăng nhập

```

CREATE PROCEDURE DeleteUserAndCharacterProfile
    @user_id INT,
    @profile_id INT
AS
BEGIN
    SET NOCOUNT ON;

    BEGIN TRY
        BEGIN TRANSACTION;

        -- Step 1: Xóa hồ sơ nhân vật
        DELETE FROM thongtinhnhavtat
        WHERE user_id = @user_id;

        -- Step 2: Xóa người dùng
        DELETE FROM nguoidung
        WHERE user_id = @user_id;
        COMMIT TRANSACTION;
    END TRY
    BEGIN CATCH
        IF @@TRANCOUNT > 0
            ROLLBACK TRANSACTION;
        THROW;
    END CATCH;
END;

CREATE TRIGGER trgDeleteUserCleanup
ON nguoidung
AFTER DELETE
AS
BEGIN
    SET NOCOUNT ON;

    DECLARE @user_id INT;

    SELECT @user_id = user_id
    FROM deleted;

    -- Xóa hồ sơ nhân vật
    DELETE FROM thongtinhnhavtat
    WHERE user_id = @user_id;

    -- Xóa nhật ký hoạt động
    DELETE FROM lichichsuhoatdong
    WHERE user_id = @user_id;
END;

```

- Chức năng check login

```

CREATE PROCEDURE CheckLogin
    @Tendangnhap NVARCHAR(50),
    @Matkhan NVARCHAR(50)
AS
BEGIN
    SET NOCOUNT ON;

    DECLARE @PasswordHashInDB NVARCHAR(255);
    SET @PasswordHashInDB = (SELECT Matkhan FROM nguoidung WHERE Tendangnhap = @Tendangnhap);

    IF @PasswordHashInDB IS NOT NULL AND @PasswordHashInDB = HASHBYTES('SHA2_256', @Matkhan)
        SELECT 'Đăng nhập thành công' AS Result;
    ELSE
        SELECT 'Đăng nhập không thành công' AS Result;
END;

```

- Báo cáo số người dùng đã tạo trong tháng

The screenshot displays the Microsoft SQL Server Enterprise Manager interface. The central pane shows a query window with the following SQL code:

```

-- Báo cáo số người dùng được tạo trong tháng
CREATE PROCEDURE CountUsersCreatedThisMonth
AS
BEGIN
    SET NOCOUNT ON;

    DECLARE @CurrentMonth INT;
    DECLARE @CurrentYear INT;

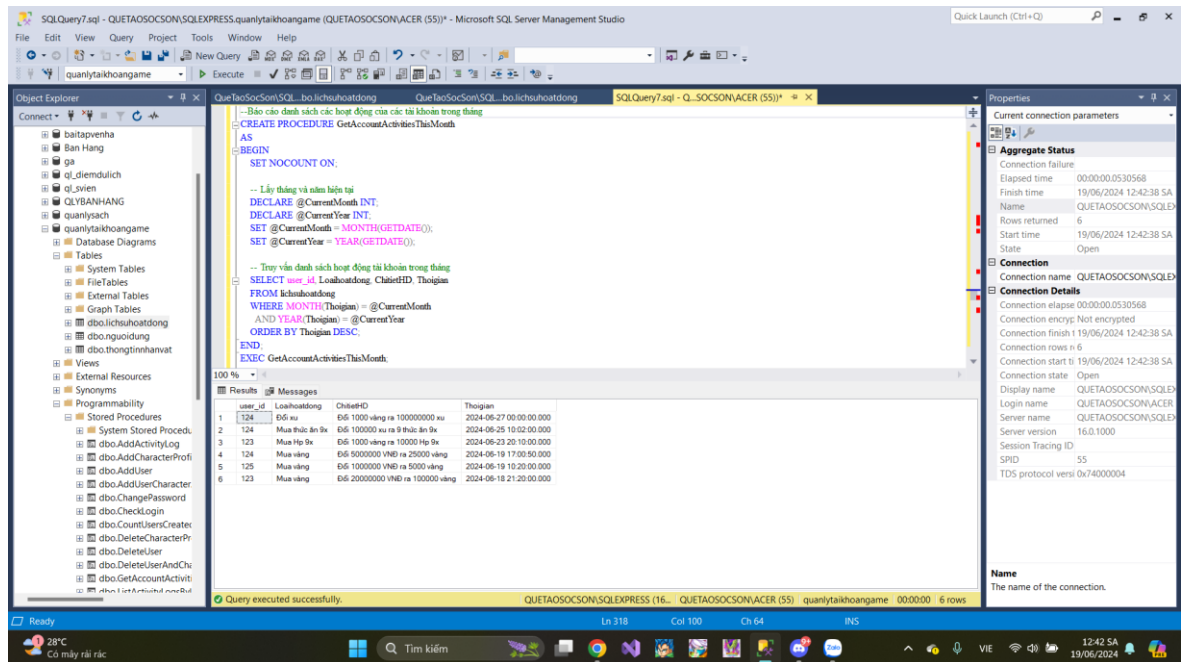
    -- Lấy tháng và năm hiện tại
    SET @CurrentMonth = MONTH(GETDATE());
    SET @CurrentYear = YEAR(GETDATE());

    -- Truy vấn để đếm số người dùng được tạo trong tháng
    SELECT COUNT(*) AS TotalUsersCreated
    FROM nguoidung
    WHERE MONTH(ngaytaoTk) = @CurrentMonth
    AND YEAR(ngaytaoTk) = @CurrentYear;
END;
EXEC CountUsersCreatedThisMonth;

```

The bottom pane shows the results of the query execution, which returned a single row with the value 2 in the 'TotalUsersCreated' column. The right-hand pane displays the 'Properties' window for the current connection, showing details such as the connection name, server name, and session ID.

- Báo cáo danh sách các hoạt động của các tài khoản trong tháng



*Cursor:

- Procedure ListUsersAndProfiles sử dụng một cursor (curUsers) để duyệt qua danh sách các người dùng trong bảng nguoidung và thông tin nhân vật tương ứng từ bảng thongtinnhanvat. Các thông tin của từng người dùng và thông tin nhân vật sẽ được in ra màn hình bằng câu lệnh PRINT.

```

DECLARE curUsers CURSOR FOR
SELECT u.user_id, u.Tendangnhap, p.TenNV, p.Capdo, p.EXP, p.Vip, p.Danhchieu, p.Tiendanap, p.Tiendachung
FROM nguoidung u
LEFT JOIN thongtinnhanvat p ON u.user_id = p.user_id;

OPEN curUsers;

FETCH NEXT FROM curUsers INTO @user_id, @Tendangnhap, @TenNV, @Capdo, @EXP, @Vip, @Danhchieu, @Tiendanap, @Tiendachung;
WHILE @@FETCH_STATUS = 0
BEGIN
    PRINT 'User ID: ' + CAST(@user_id AS NVARCHAR(10)) + ', Username: ' + @Tendangnhap;
    PRINT 'Character Name: ' + @TenNV + ', Level: ' + CAST(@Capdo AS NVARCHAR(10)) + ', EXP: ' + CAST(@EXP AS NVARCHAR(20));
    PRINT 'VIP Level: ' + CAST(@Vip AS NVARCHAR(10)) + ', Title: ' + @Danhchieu;
    PRINT 'Deposit: ' + @Tiendanap + ', Usage: ' + @Tiendachung;
    PRINT '-----';

    FETCH NEXT FROM curUsers INTO @user_id, @Tendangnhap, @TenNV, @Capdo, @EXP, @Vip, @Danhchieu, @Tiendanap, @Tiendachung;
END

CLOSE curUsers;
DEALLOCATE curUsers;
END;

```

*Trigger:

- Trigger sẽ được kích hoạt sau khi một người dùng (nguoidung) được xóa. Khi người dùng bị xóa, trigger này sẽ tự động xóa thông tin nhân vật (character profile) của người dùng đó từ bảng thongtinnhanvat và lịch sử hoạt động (activity log) của người dùng đó từ bảng

lichlichsuhoatdong.

```
CREATE TRIGGER trgDeleteUserCleanup
ON nguoidung
AFTER DELETE
AS
BEGIN
    SET NOCOUNT ON;

    DECLARE @user_id INT;

    SELECT @user_id = user_id
    FROM deleted;

    -- Xóa hồ sơ nhân vật
    DELETE FROM thongtinhnhanvat
    WHERE user_id = @user_id;

    -- Xóa nhật ký hoạt động
    DELETE FROM lichlichsuhoatdong
    WHERE user_id = @user_id;

END;
```

- Trigger trgUpdateNgaytaoTk sẽ được kích hoạt sau khi có sự kiện INSERT vào bảng nguoidung. Nó lấy ngày giờ hiện tại bằng GETDATE() và cập nhật giá trị ngaytaoTk cho người dùng vừa được thêm vào.

```
CREATE TRIGGER trgUpdateNgaytaoTk
ON nguoidung
AFTER INSERT
AS
BEGIN
    SET NOCOUNT ON;

    DECLARE @user_id INT;
    DECLARE @now DATETIME;

    -- Lấy ngày giờ hiện tại
    SET @now = GETDATE();

    -- Cập nhật ngaytaoTk cho người dùng mới được chèn
    SELECT @user_id = user_id FROM inserted;

    UPDATE nguoidung
    SET ngaytaoTk = @now
    WHERE user_id = @user_id;

END;
```

The screenshot shows the Microsoft SQL Server Enterprise Manager interface. The left pane displays the Object Explorer with the database structure. The center pane shows a query window with the following SQL code:

```
-- Họ Tên: Trần Chiến Thắng
-- Ngày làm: 18/06/2024 11h30
-- Mô tả: SP này làm về quản lý tài khoản game

CREATE PROCEDURE AddUser
    @Tendangup NVARCHAR(100),
    @Matkhau NVARCHAR(255),
    @Email NVARCHAR(100)
AS
BEGIN
    SET NOCOUNT ON;

    DECLARE @PasswordHash NVARCHAR(255);
    SET @PasswordHash = HASHBYTES(SHA2_256, @Matkhau);

    INSERT INTO nguoidung (Tendangup, Matkhau, Email)
    VALUES (@Tendangup, @PasswordHash, @Email);
END;

CREATE PROCEDURE DeleteUser
    @user_id INT
AS
BEGIN
    SET NOCOUNT ON;

    DELETE FROM nguoidung
    WHERE user_id = @user_id;
END;
```

The right pane shows the Results window with the following data:

log_id	user_id	Loaishoatdong	Chisethoi	Thoigian
1	123	Mua vàng	66 20000000 VNĐ ra 100000 vàng	2024-06-18 21:20:00.000
2	124	Mua vàng	66 5000000 VNĐ ra 250000 vàng	2024-06-19 17:00:50.000
3	125	Mua vàng	66 1000000 VNĐ ra 50000 vàng	2024-06-19 10:20:00.000
4	123	Mua Hp Dx	66 1000 vàng ra 10000 Hp Dx	2024-06-23 20:10:00.000
5	124	Mua thức ăn Dx	66 100000 xu ra 9 thức ăn Dx	2024-06-25 10:02:00.000
6	124	Đổi xu	66 1000 vàng ra 100000000 xu	2024-06-27 00:00:00.000

The bottom status bar indicates the query was executed successfully, returning 6 rows.

