

# Project Report + EC

## Group Information

Group 61: Danin Namiranian and Siddharta Dutta

## Implementation Description

Include screenshots and/or code snippets for each query. In addition, explain how you implemented each query and what purpose it fulfills. (no more than 1-2 sentences per query)

Our project involves creating a rental management system for a game rental store. The system is designed to handle user authentication, profile management, game catalog browsing, favorite game management, rental order processing, and tracking information updates. We implemented the database schema using PostgreSQL and developed a Java application to interact with the database, providing a user-friendly interface for various operations.

### Authentication Queries

#### Sign Up

```
String query = String.format(
    "INSERT INTO Users (login, password, role, favGames, phoneNum, numOverDueGames) " +
    "VALUES ('%s', '%s', '%s', '%s', '%s', %d)",
    login, password, role, favGames, phoneNum, numOverDueGames);
```

*Purpose:* Registers a new user by inserting their details into the `Users` table

#### Sign In:

```
String query = String.format( "SELECT login FROM Users WHERE login = '%s' AND password = '%s'", login,
password);
```

*Purpose:* Verifies user credentials for authentication

### Profile Management Queries

#### View Profile:

```
String query = String.format(
    "SELECT login, password, role, favGames, phoneNum, numOverDueGames FROM Users WHERE login = '%s'",
    authorisedUser);
```

*Purpose:* Retrieves and displays the profile information of the logged-in user.

#### Edit Profile:

```
String updateQuery = String.format(
    "UPDATE Users SET password = '%s', phoneNum = '%s' WHERE login = '%s'",
    newPassword, newPhoneNum, authorisedUser);
```

*Purpose:* Updates the profile information of the logged-in user.

### Game Management Queries

#### View All Games:

```
String query = "SELECT gameID, price, genre, gameName, description FROM Catalog";
```

*Purpose:* Retrieves and displays all games from the catalog.

#### View Favorite Games:

```
String query = String.format(
    "SELECT gameID, price, genre, gameName, description " +
    "FROM Catalog WHERE gameID IN (SELECT unnest(string_to_array(favGames, ',')) " +
    "FROM Users WHERE login = '%s')",
    authorisedUser);
```

*Purpose:* Retrieves and displays the favorite games of the logged-in user.

#### Add Favorite Game:

```
String updateQuery = String.format(
    "UPDATE Users SET favGames = '%s' WHERE login = '%s'",
    favGames, authorisedUser);
```

*Purpose:* Adds a game to the favorite games list of the logged-in user.

#### Remove Favorite Game:

```
String updateQuery = String.format(
    "UPDATE Users SET favGames = '%s' WHERE login = '%s'",
    favGames, authorisedUser);
```

*Purpose:* Removes a game from the favorite games list of the logged-in user.

## Rental Management Queries

#### Create New Order:

```
String insertOrderQuery = String.format(
    "INSERT INTO RentalOrder (rentalOrderID, login, noOfGames, totalPrice, orderTimestamp, dueDate) " +
    "VALUES ('%s', '%s', %d, %f, NOW(), NOW() + INTERVAL '7 days')",
    rentalOrderID, authorisedUser, unitsOrdered, totalPrice);
```

*Purpose:* Creates a new rental order for the logged-in user.

#### View All Orders:

```
String query = String.format(
    "SELECT rentalOrderID, noOfGames, totalPrice, orderTimestamp, dueDate " +
    "FROM RentalOrder WHERE login = '%s'",
    authorisedUser);
```

*Purpose:* Retrieves and displays all rental orders of the logged-in user.

#### View Recent Orders:

```
String query = String.format(
    "SELECT rentalOrderID, noOfGames, totalPrice, orderTimestamp, dueDate " +
    "FROM RentalOrder WHERE login = '%s' ORDER BY orderTimestamp DESC LIMIT 5",
    authorisedUser);
```

*Purpose:* Retrieves and displays the recent rental orders of the logged-in user.

#### View Order Details:

```
String rentalOrderQuery = String.format(
    "SELECT rentalOrderID, noOfGames, totalPrice, orderTimestamp, dueDate " +
    "FROM RentalOrder WHERE rentalOrderID = '%s'",
    rentalOrderID);
```

*Purpose:* Retrieves and displays detailed information about a specific rental order.

## Admin Management Queries

#### Update User:

```
String updateQuery = String.format(
    "UPDATE Users SET password = '%s', role = '%s', phoneNum = '%s', favGames = '%s' WHERE login = '%s'",
    newPassword, newRole, newPhoneNum, newFavGames, userLogin);
```

*Purpose:* Allows a manager to update user information.

**Update Catalog:**

```
String updateQuery = String.format(
    "UPDATE Catalog SET gameName = '%s', genre = '%s', price = '%s', description = '%s', imageURL = '%s'
    WHERE gameID = '%s'",
    newGameName, newGenre, newPrice, newDescription, newImageURL, gameID);
```

*Purpose:* Allows a manager to update game information in the catalog.

**Update Tracking Info:**

```
String updateQuery = String.format(
    "UPDATE TrackingInfo SET status = '%s', currentLocation = '%s', courierName = '%s', additionalComments =
    '%s', lastUpdateDate = NOW() WHERE trackingID = '%s'",
    newStatus, newCurrentLocation, newCourierName, newAdditionalComments, trackingID);
```

*Purpose:* Allows a manager or employee to update tracking information.

**Problems/Findings**

During the implementation, we encountered challenges related to ensuring data consistency and integrity, particularly with cascading deletes and maintaining relational integrity across multiple tables. We also had to carefully handle user inputs to prevent SQL injection and other security vulnerabilities. Additionally, managing transactions to ensure atomicity of operations like creating orders and tracking information was a key consideration.

**Contributions**

**Danin Namiranian :**

- Designing the database schema.
- Implementing user authentication (sign up and sign in).
- Developing profile management features (view and edit profile).
- Creating game management functionalities (view all games, view favorite games, add favorite game, remove favorite game).
- Implementing order management processes (create new order, view all orders, view recent orders, view order details).
- Developing admin functionalities (update user information, update catalog information, update tracking information).
- Ensuring data consistency and integrity across multiple tables.
- Handling user inputs securely to prevent SQL injection and other vulnerabilities.
- Managing transactions to ensure atomicity of operations.
- Optimizing SQL queries for better performance.
- Developing the Java application to interact with the database and provide a user-friendly interface for various operations.

**Siddharta Dutta :**

Extra Credit

Extra Credit 1 ( HERE ARE SOME EXAMPLE )

Good User Interface A good user interface will bring more users to your application, and also more points in this phase.:

- Easy for users to explore features;
- Robust in exceptional situations, like unexpected inputs;
- Graphic interface supporting all required features

<div>Register</div> <div><div><div><div>🎮</div><div>Welcome to GameRoom</div></div></div><div><div>[01] → Sign Up</div><div>[02] → Sign In</div><div>[09] → Exit</div><div>[00] → debug</div></div><div><div>👉</div><div>Select an option: 1</div></div><div><div>-----</div><div><div>🔒</div><div>Create Account   Let's get you started</div></div><div>-----</div></div><div><div><div>👤</div><div>Username: Danin</div></div><div><div>🔒</div><div>Password: Pass123</div></div><div><div>📞</div><div>PhoneNum: 9493573929</div></div></div><div><div>-----</div><div><div>✅</div><div>Your Account has been created</div></div><div>-----</div></div><div><div>Press [ENTER] to continue:</div><div></div></div></div>
---



View Games Sorted by Price

> game0388	\$19.99	Platform	🐉 LittleBigPlanet 2   Platform: PS3; Publisher: Sony Computer Entertainment
> game0089	\$18.99	Role-Playing	🐉 Pokemon Platinum Version   Platform: DS; Publisher: Nintendo
> game0043	\$18.99	Racing	🐉 Mario Kart 7   Platform: 3DS; Publisher: Nintendo
> game0453	\$18.99	Sports	🐉 World Class Track Meet   Platform: NES; Publisher: Namco Bandai Games
> game0367	\$18.99	Platform	🐉 Yoshi's Island DS   Platform: DS; Publisher: Nintendo
> game0269	\$18.99	Misc	🐉 Mario Paint   Platform: SNES; Publisher: Nintendo
> game0207	\$18.99	Shooter	🐉 Link's Crossbow Training   Platform: Wii; Publisher: Nintendo
> game0116	\$18.99	Action	🐉 Uncharted 3: Drake's Deception   Platform: PS3; Publisher: Sony Computer Entertainment
> game0161	\$18.99	Racing	🐉 Forza Motorsport 3   Platform: X360; Publisher: Microsoft Game Studios
> game0173	\$18.99	Racing	🐉 Need for Speed: Shift   Platform: PS3; Publisher: Electronic Arts
> game0287	\$18.99	Misc	🐉 Carnival Games   Platform: Wii; Publisher: Take-Two Interactive
> game0106	\$18.99	Fighting	🐉 Tekken 3   Platform: PS; Publisher: Sony Computer Entertainment
> game0101	\$18.99	Action	🐉 The Legend of Zelda: Twilight Princess   Platform: Wii; Publisher: Nintendo
> game0107	\$18.99	Platform	🐉 Crash Bandicoot 3: Warped   Platform: PS; Publisher: Sony Computer Entertainment
> game0036	\$17.99	Sports	🐉 Rocket League   Platform: Multi; Publisher: Epic Games
> game0110	\$17.99	Role-Playing	🐉 Fallout 4   Platform: PS4; Publisher: Bethesda Softworks
> game0396	\$17.99	Action	🐉 Metal Gear Solid V: The Phantom Pain   Platform: PS4; Publisher: Konami Digital Entertainment
> game0309	\$17.99	Role-Playing	🐉 Monster Hunter 4 Ultimate   Platform: 3DS; Publisher: Nintendo
> game0446	\$17.99	Strategy	🐉 Command & Conquer: Red Alert   Platform: PC; Publisher: Virgin Interactive
> game0196	\$17.99	Misc	🐉 Guitar Hero II   Platform: PS2; Publisher: RedOctane
> game0356	\$17.99	Simulation	🐉 Cooking Mama 2: Dinner With Friends   Platform: DS; Publisher: 505 Games
> game0383	\$17.99	Shooter	🐉 Far Cry 3   Platform: PS3; Publisher: Ubisoft
> game0403	\$17.99	Misc	🐉 English Training: Have Fun Improving Your Skills!   Platform: DS; Publisher: Nintendo
> game0447	\$17.99	Role-Playing	🐉 Dragon Warrior IV   Platform: NES; Publisher: Enix Corporation
> game0280	\$17.99	Sports	🐉 Madden NFL 2003   Platform: PS2; Publisher: Electronic Arts
> game0213	\$17.99	Misc	🐉 Mario Party   Platform: N64; Publisher: Nintendo
> game0230	\$17.99	Action	🐉 Tomb Raider   Platform: PS; Publisher: Eidos Interactive
> game0495	\$17.99	Shooter	🐉 Titanfall   Platform: XOne; Publisher: Electronic Arts
> game0496	\$16.99	Misc	🐉 Sonic Mega Collection Plus   Platform: PS2; Publisher: Sega
> game0067	\$16.99	Role-Playing	🐉 Final Fantasy VII   Platform: PS; Publisher: Sony Computer Entertainment
> game0468	\$16.99	Shooter	🐉 Gears of War: Ultimate Edition   Platform: XOne; Publisher: Microsoft Game Studios
> game0052	\$16.99	Platform	🐉 Super Mario Bros.: The Lost Levels   Platform: NES; Publisher: Nintendo
> game0253	\$16.99	Racing	🐉 Need for Speed: Most Wanted   Platform: PS2; Publisher: Electronic Arts
> game0421	\$16.99	Shooter	🐉 007: Tomorrow Never Dies   Platform: PS; Publisher: Electronic Arts
> game0127	\$16.99	Misc	🐉 Guitar Hero: World Tour   Platform: Wii; Publisher: Activision
> game0226	\$16.99	Sports	🐉 Tony Hawk's Pro Skater 2   Platform: PS; Publisher: Activision
> game0369	\$16.99	Platform	🐉 Spyro 2: Ripto's Rage!   Platform: PS; Publisher: Sony Computer Entertainment
> game0456	\$16.99	Platform	🐉 Donkey Kong   Platform: GB; Publisher: Nintendo
> game0457	\$16.99	Simulation	🐉 The Sims: Vacation   Platform: PC; Publisher: Electronic Arts
> game0368	\$16.99	Action	🐉 Rugrats in Paris: The Movie   Platform: PS; Publisher: THQ
> game0277	\$16.99	Action	🐉 Teenage Mutant Ninja Turtles   Platform: NES; Publisher: Palcom
> game0407	\$16.99	Action	🐉 Star Wars Episode III: Revenge of the Sith   Platform: PS2; Publisher: LucasArts
> game0233	\$15.99	Action	🐉 The Legend of Zelda: The Wind Waker   Platform: GC; Publisher: Nintendo
> game0498	\$15.99	Sports	🐉 World Soccer Winning Eleven 7 International   Platform: PS2; Publisher: Konami Digital Entertainment
> game0490	\$15.99	Shooter	🐉 Valorant   Platform: Multi; Publisher: Riot Games
> game0462	\$15.99	Action	🐉 Saints Row 2   Platform: X360; Publisher: THQ
> game0189	\$15.99	Simulation	🐉 Tomodachi Life   Platform: 3DS; Publisher: Nintendo
> game0348	\$15.99	Puzzle	🐉 Professor Layton and the Last Specter   Platform: DS; Publisher: Nintendo
> game0268	\$15.99	Strategy	🐉 Warcraft II: Tides of Darkness   Platform: PC; Publisher: Activision
> game0083	\$15.99	Action	🐉 FIFA Soccer 13   Platform: PS3; Publisher: Electronic Arts
> game0029	\$15.99	Racing	🐉 Gran Turismo 3: A-Spec   Platform: PS2; Publisher: Sony Computer Entertainment
> game0240	\$15.99	Platform	🐉 Pitfall!   Platform: 2600; Publisher: Activision
> game0426	\$15.99	Action	🐉 God of War: Chains of Olympus   Platform: PSP; Publisher: Sony Computer Entertainment

View Games Filter by Price

♦ Enter Maximum Price: 17			
> game0029	\$15.99	Racing	🐉 Gran Turismo 3: A-Spec   Platform: PS2; Publisher: Sony Computer Entertainment
> game0052	\$16.99	Platform	🐉 Super Mario Bros.: The Lost Levels   Platform: NES; Publisher: Nintendo
> game0067	\$16.99	Role-Playing	🐉 Final Fantasy VII   Platform: PS; Publisher: Sony Computer Entertainment
> game0083	\$15.99	Action	🐉 FIFA Soccer 13   Platform: PS3; Publisher: Electronic Arts
> game0127	\$16.99	Misc	🐉 Guitar Hero: World Tour   Platform: Wii; Publisher: Activision
> game0189	\$15.99	Simulation	🐉 Tomodachi Life   Platform: 3DS; Publisher: Nintendo
> game0226	\$16.99	Sports	🐉 Tony Hawk's Pro Skater 2   Platform: PS; Publisher: Activision
> game0233	\$15.99	Action	🐉 The Legend of Zelda: The Wind Waker   Platform: GC; Publisher: Nintendo
> game0240	\$15.99	Platform	🐉 Pitfall!   Platform: 2600; Publisher: Activision
> game0253	\$16.99	Racing	🐉 Need for Speed: Most Wanted   Platform: PS2; Publisher: Electronic Arts
> game0268	\$15.99	Strategy	🐉 Warcraft II: Tides of Darkness   Platform: PC; Publisher: Activision
> game0277	\$16.99	Action	🐉 Teenage Mutant Ninja Turtles   Platform: NES; Publisher: Palcom
> game0348	\$15.99	Puzzle	🐉 Professor Layton and the Last Specter   Platform: DS; Publisher: Nintendo
> game0368	\$16.99	Action	🐉 Rugrats in Paris: The Movie   Platform: PS; Publisher: THQ
> game0369	\$16.99	Platform	🐉 Spyro 2: Ripto's Rage!   Platform: PS; Publisher: Sony Computer Entertainment
> game0407	\$16.99	Action	🐉 Star Wars Episode III: Revenge of the Sith   Platform: PS2; Publisher: LucasArts
> game0421	\$16.99	Shooter	🐉 007: Tomorrow Never Dies   Platform: PS; Publisher: Electronic Arts
> game0426	\$15.99	Action	🐉 God of War: Chains of Olympus   Platform: PSP; Publisher: Sony Computer Entertainment
> game0456	\$16.99	Platform	🐉 Donkey Kong   Platform: GB; Publisher: Nintendo
> game0457	\$16.99	Simulation	🐉 The Sims: Vacation   Platform: PC; Publisher: Electronic Arts
> game0462	\$15.99	Action	🐉 Saints Row 2   Platform: X360; Publisher: THQ
> game0468	\$16.99	Shooter	🐉 Gears of War: Ultimate Edition   Platform: XOne; Publisher: Microsoft Game Studios
> game0490	\$15.99	Shooter	🐉 Valorant   Platform: Multi; Publisher: Riot Games
> game0496	\$16.99	Misc	🐉 Sonic Mega Collection Plus   Platform: PS2; Publisher: Sega
> game0498	\$15.99	Sports	🐉 World Soccer Winning Eleven 7 International   Platform: PS2; Publisher: Konami Digital Entertainment
Press [ENTER] to continue: <input type="text"/>			

## Extra Credit 3 ( Performance Tuning )

The performance can be improved if indexes are used properly. You will receive points only if the index will actually be used in your queries. You should defend why you have chosen to use an index at some particular table. For your submission, you should put index declarations in project/sql/src/create\_indexes.sql. Points will also be awarded for efficient queries (for instance, using appropriate joins, or using IF EXISTS functionality).

```
-- Drop existing indexes if they exist
DROP INDEX IF EXISTS idx_users_phoneNum;
DROP INDEX IF EXISTS idx_catalog_gameName;
DROP INDEX IF EXISTS idx_catalog_genre;
DROP INDEX IF EXISTS idx_rentalOrder_login;
DROP INDEX IF EXISTS idx_trackingInfo_rentalOrderID;
DROP INDEX IF EXISTS idx_gamesInOrder_rentalOrderID;
DROP INDEX IF EXISTS idx_gamesInOrder_gameID;

-- Indexes for Users table
CREATE INDEX idx_users_phoneNum ON Users(phoneNum);

-- Indexes for Catalog table
CREATE INDEX idx_catalog_gameName ON Catalog(gameName);
CREATE INDEX idx_catalog_genre ON Catalog(genre);

-- Indexes for RentalOrder table
CREATE INDEX idx_rentalOrder_login ON RentalOrder(login);

-- Indexes for TrackingInfo table
CREATE INDEX idx_trackingInfo_rentalOrderID ON TrackingInfo(rentalOrderID);

-- Indexes for GamesInOrder table
CREATE INDEX idx_gamesInOrder_rentalOrderID ON GamesInOrder(rentalOrderID);
CREATE INDEX idx_gamesInOrder_gameID ON GamesInOrder(gameID);
```

### Defending my Index Choices

- 1. **Users Table:**
  - **PhoneNum:** The `idx_users_phoneNum` index is beneficial for the `profile_edit` function, which may search users by phone number to update their profile.
- 2. **Catalog Table:**
  - **GameName and Genre:**
    - `idx_catalog_gameName` is used in the `games_view` function for filtering games by name.
    - `idx_catalog_genre` is used in the `games_view` function for filtering games by genre.
- 3. **RentalOrder Table:**
  - **Login:** The `idx_rentalOrder_login` index improves the performance of `order_view_all`, `order_view_recent`, and `order_push` functions, which frequently query orders by user login.
- 4. **TrackingInfo Table:**
  - **RentalOrderID:** The `idx_trackingInfo_rentalOrderID` index is crucial for the `order_view_info` function to fetch tracking information based on the rental order ID.
- 5. **GamesInOrder Table:**
  - **RentalOrderID and GameID:**
    - `idx_gamesInOrder_rentalOrderID` is used in the `order_push` and `order_view_info` functions to join and retrieve games for specific orders.
    - `idx_gamesInOrder_gameID` enhances performance for queries checking game orders in functions like `games_view_fav` and `games_push_fav`.

Extra Credit 4 ( Any Other Fancy Stuff... )

Please feel free to show any of your ideas to improve your project! The only thing required will be clear documentation in your report!

1. Dynamic Dashboard System

- **Description:** I created a dynamic dashboard system that displays different menus based on the user's role and available functionalities. This makes the interface interactive and user-friendly.
- **Implementation:**
  - Functions like `menu_auth()` and `menu_dash(String username)` display various options based on the user's authentication status and role.
  - I use `print_option()` and `print_title()` methods to format and present menu options with emojis for visual appeal.
  - The dashboard displays user-specific sections like profile, games, and orders, providing a personalized experience

```
♦ Enter Rental Order ID: : gamerentalorder2499

-----

📦 Rental Order Information:
-----

ID Rental Order ID: gamerentalorder2499
🎮 Number of Games: 3
💰 Total Price: $140.97
📅 Order Timestamp: 2013-11-13 05:58:54
📅 Due Date: 2013-12-13 00:00:00

-----

🎮 Game Information:
-----

ID Game ID: game0325
🎮 Game Name: LEGO Indiana Jones: The Original Adventures
🎮 Genre: Action
💰 Price: $46.99
📖 Description: Platform: X360; Publisher: Activision

-----

🚚 Tracking Information:
-----

ID Tracking ID: trackingid2499
📦 Status: Out for Delivery
📍 Current Location: Philadelphia,PA
🚚 Courier Name: USPS
📅 Last Update Date: 2013-11-19 05:58:54
📄 Additional Comments:

Press [ENTER] to continue: 
```

2. User-Friendly Error Handling

- **Description:** To enhance the user experience, I provide detailed error messages and user feedback.
- **Implementation:**
  - I catch exceptions and display user-friendly messages using `print_message()`.
  - I also clean up error messages by removing extra spaces with `trim()` and `replaceAll()` methods, making them more readable.

```
🎮 Welcome to GameRoom

[01] → Sign Up
[02] → Sign In
[09] → Exit
[00] → debug

👉 Select an option: 1

-----

🔒 Create Account | Let's get you started
-----

👤 Username: Danin
🔒 Password: 1
📞 PhoneNum: 1

-----

❌ ERROR: duplicate key value violates unique constraint "users_pkey"
-----

Press [ENTER] to continue: 
```

```
🎮 Welcome to GameRoom

[01] → Sign Up
[02] → Sign In
[09] → Exit
[00] → debug

👉 Select an option: 2

-----

🔒 Login Account | Manage your games and more
-----

👤 Username: Danin
🔒 Password: 555

-----

❌ Invalid login or password.
-----

Press [ENTER] to continue: 
```

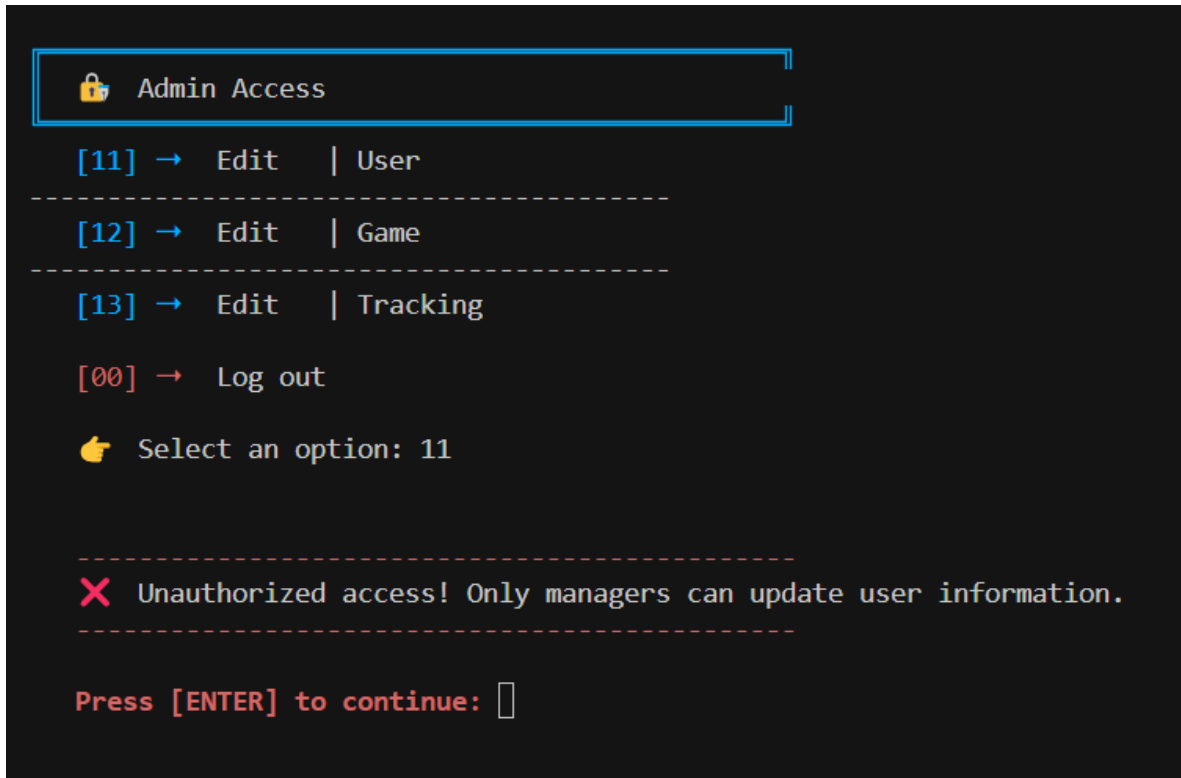
### 3. Session Management

- **Description:** I maintain user sessions and display personalized messages to enhance user engagement.
- **Implementation:**
  - Functions like `menu_dash(String username)` greet users with a personalized welcome message based on their login credentials.



### 4. Role-Based Access Control

- **Description:** I've implemented role-based access control to restrict certain functionalities to authorized users.
- **Implementation:**
  - Functions like `admin_update_user()`, `admin_update_catalog()`, and `admin_update_trackingInfo()` check the user's role before allowing access to update functionalities.





## 5. Advanced Data Display

- **Description:** I display data in a structured format for better readability and user comprehension.
- **Implementation:**
  - Functions such as `auth_all()`, `games_view()`, and `order_view_info()` use ASCII tables to present user, game, and order information in a clear and organized manner.

```
👉 Select an option: 8

> gamerentalorder1551 | 📦 3 | $83.97 | ➡ 2013-07-11 18:50:35 | 🚚 2013-08-10 00:00:00
> gamerentalorder2026 | 📦 5 | $229.95 | ➡ 2013-01-10 06:10:55 | 🚚 2013-02-09 00:00:00
> gamerentalorder2315 | 📦 3 | $104.97 | ➡ 2017-09-21 03:57:32 | 🚚 2017-10-21 00:00:00
> gamerentalorder2499 | 📦 3 | $140.97 | ➡ 2013-11-13 05:58:54 | 🚚 2013-12-13 00:00:00
> gamerentalorder2915 | 📦 4 | $183.96 | ➡ 2015-02-15 03:17:12 | 🚚 2015-03-17 00:00:00
> gamerentalorder3277 | 📦 4 | $163.96 | ➡ 2012-09-28 11:07:28 | 🚚 2012-10-28 00:00:00
> gamerentalorder3495 | 📦 2 | $95.98 | ➡ 2014-09-29 01:34:53 | 🚚 2014-10-29 00:00:00
> gamerentalorder3689 | 📦 5 | $244.95 | ➡ 2018-10-22 00:37:02 | 🚚 2018-11-21 00:00:00
> gamerentalorder3893 | 📦 2 | $81.98 | ➡ 2021-03-28 06:15:01 | 🚚 2021-04-27 00:00:00

Press [ENTER] to continue: 
```

## 6. SQL Injection Prevention

- **Description:** I've taken measures to prevent SQL injection by properly formatting SQL queries and sanitizing user inputs.
- **Implementation:**
  - In functions like `auth_signup()` and `auth_signin()`, I use `String.format` to construct SQL queries, ensuring that user inputs are correctly handled and potential threats are mitigated.

## 7. UUID Generation for Unique IDs

- **Description:** I've implemented the use of UUIDs to generate unique identifiers for rental orders and tracking information.
- **Implementation:**
  - In functions like `order_push()`, I use `UUID.randomUUID().toString()` to create unique `rentalOrderID` and `trackingID`.