

## Instructions for Team Members - Setting Up Admin Profile and Reports Table

To ensure everyone has the correct setup, follow these **essential steps** after pulling the latest changes from the repository.

---

### 1. Database Setup

You need to **update the database schema** to support:

- **Admin Profile Picture Uploads**
- **Moderation Reports Table**

#### A. Modify the **users** Table

Run this SQL command to add a **profile\_picture** column:

```
ALTER TABLE users ADD COLUMN profile_picture VARCHAR(255) DEFAULT  
'/uploads/default-avatar.png';
```

#### B. Ensure the **reports** Table Exists

If the **reports** table is missing, create it by running:

```
CREATE TABLE reports (  
    report_id SERIAL PRIMARY KEY,  
    reported_user_id INT NOT NULL,  
    reported_by_user_id INT NOT NULL,  
    reason TEXT NOT NULL,  
    status VARCHAR(20) CHECK (status IN ('Pending', 'Reviewed',  
'Action Taken')) DEFAULT 'Pending',  
    created_at TIMESTAMP DEFAULT NOW(),  
    FOREIGN KEY (reported_user_id) REFERENCES users(user_id) ON DELETE  
CASCADE,  
    FOREIGN KEY (reported_by_user_id) REFERENCES users(user_id) ON  
DELETE CASCADE  
);
```

## C. Verify Data Exists

Run this to check if the reports table has any entries:

```
SELECT * FROM reports;
```

If empty, manually insert test data:

```
INSERT INTO reports (reported_user_id, reported_by_user_id, reason,
status)
VALUES (1, 2, 'Toxic behavior in chat', 'Pending');
```

---

## 2. Install Required Dependencies

Since the new features involve **file uploads** and path handling, install the following dependencies:

```
npm install express multer path fs
```

These packages are **essential** for handling profile picture uploads.

---

## 3. Ensure the **uploads** Directory Exists

Make sure your **project directory** contains an **uploads** folder inside **public/** for storing profile pictures:

```
mkdir -p public/uploads
```

---

## 4. Start the Server

Run the application to check if everything is working:

```
npm start
```

---

## 5. Expected Features After Setup

### ✓ Admin Profile Page

- Admins can update their **username, email, and password**.
- **Profile picture upload support** (stored in `/uploads/`).
- Displays **moderation reports** on toxic behavior.

### ✓ Navbar Updates

- Both **Gamers and Admins** will have their profile picture displayed in the navbar when logged in.
- 

## 6. Install multer using your vs code terminal for profile pic upload:

```
npm install express multer
```

### Final Note:

Once you complete these steps, you should be able to test the Admin features. Let me know if you face any issues. Do not push `.env` files or the `uploads/` directory to GitHub.