**Project Name: Modern Connect**

**Modern Connect** ek college management system hai jisme **Admin**, **Teacher**, aur **Student** teen types ke users hote hain. Har user ka role alag hota hai aur unka kaam system mein bhi alag hota hai.

**Technologies Used:**

1. **Front-end:**
   * **HTML:** Website structure design karne ke liye.
   * **CSS:** Website ko visually appealing banane ke liye (layout aur design).
   * **JavaScript:** User interactions, forms validation, aur interactive elements handle karne ke liye.
   * **Bootstrap:** Responsive design ke liye, jisse website har device (mobile ya desktop) par perfectly dikhaye.
2. **Back-end:**
   * **PHP:** Server-side programming language, jo login, data management (fetching, updating), aur CRUD operations ko handle karta hai.
3. **Database:**
   * **PostgreSQL:** Relational database management system, jisme hum user data, courses, aur other essential records ko securely store karte hain.

**Roles and Responsibilities:**

1. **Admin Role:**
   * Admin ko sabhi tasks ka **control** hota hai.
   * Admin **CRUD operations** karta hai, jaise ki:
     + Naye **students**, **teachers**, **courses** aur **departments** ko add/update/delete karna.
   * **Assigning Teachers to Courses:** Admin decide karta hai ki kaunse teacher ko kis course ka assignment diya jayega.
2. **Teacher Role:**
   * Teachers apne **assigned courses** ko manage karte hain.
   * Teachers **assignments aur notes** share karte hain apne students ke liye.
   * Teachers **students ke progress** ko monitor karte hain aur zaroori notifications aur announcements bhejte hain.
3. **Student Role:**
   * Students apne **courses aur assignments** ko dekhte hain aur unhe submit karte hain.
   * Students ko **teachers ke notes** milte hain aur woh **announcements** dekh sakte hain.

**How it Works:**

1. **Admin:**
   * Admin login karke sabhi users ko manage karta hai, jaise ki students, teachers, aur courses ko add/edit/assign karna.
   * **Admin ka kaam:** CRUD operations, courses ka assignment, aur system monitor karna.
2. **Teacher:**
   * Teacher apne login ke baad assigned courses aur students ko dekhte hain.
   * Teachers apne students ke assignments aur notes share karte hain aur unki progress monitor karte hain.
3. **Student:**
   * Students apne courses aur assignments dekh kar unhe submit karte hain.
   * Students apne teachers ke notes aur updates ko follow karte hain.

**Advantages:**

1. **Easy Management:** Har role ke liye specific tasks define hain, jisse system efficiently chal raha hota hai.
2. **Efficient Communication:** Teachers aur students ke beech easy aur quick communication hoti hai.
3. **Time-Saving:** Manual kaam kaafi reduce ho jata hai, sab kuch ek organized system ke andar automate hota hai.
4. **Security:** Role-based access control hota hai, jisme har user ko unke specific role ke hisaab se data access milta hai.

**Disadvantages:**

1. **Internet Dependency:** Agar internet connection nahi hai, toh system ka access nahi milega.
2. **Training:** Users ko thoda time dena padta hai system samajhne aur use karne ke liye.

**Conclusion:**

**Modern Connect** ka main goal college ke daily tasks ko **automate** aur **simplify** karna hai, taki **Admin**, **Teacher**, aur **Student** ka kaam easily aur efficiently ho sake. Is project ke through, users ko ek better, secure aur organized platform milta hai apne daily tasks ke liye. Har user ka role clearly defined hai aur system highly **user-friendly** hai.

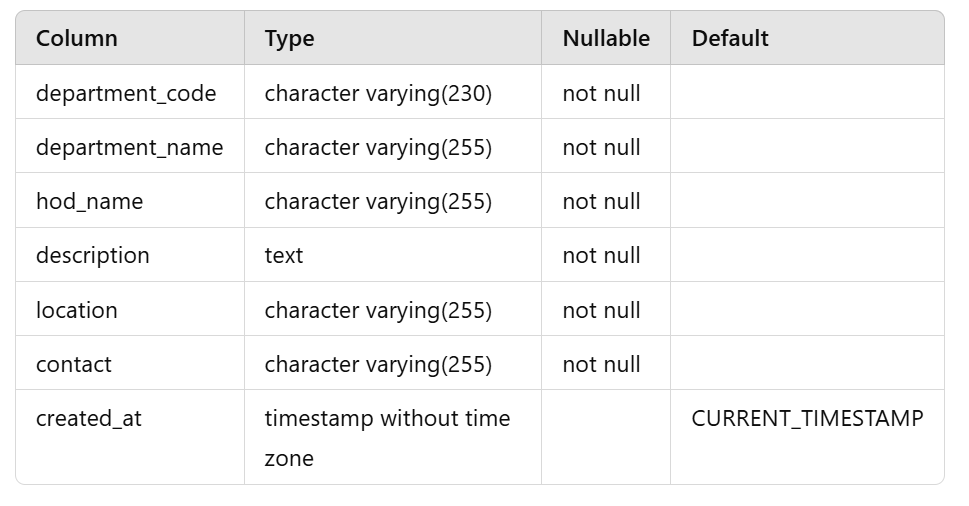
**Summary:**

* Ye project **major** aur practical project hai, jo **college management** ko improve karta hai.
* Aapne technology stack ko achhe se mention kiya hai, jaise **front-end (HTML, CSS, JS, Bootstrap)**, **back-end (PHP)** aur **database (PostgreSQL)**, jo real-world implementation ke liye bilkul sahi choices hain.

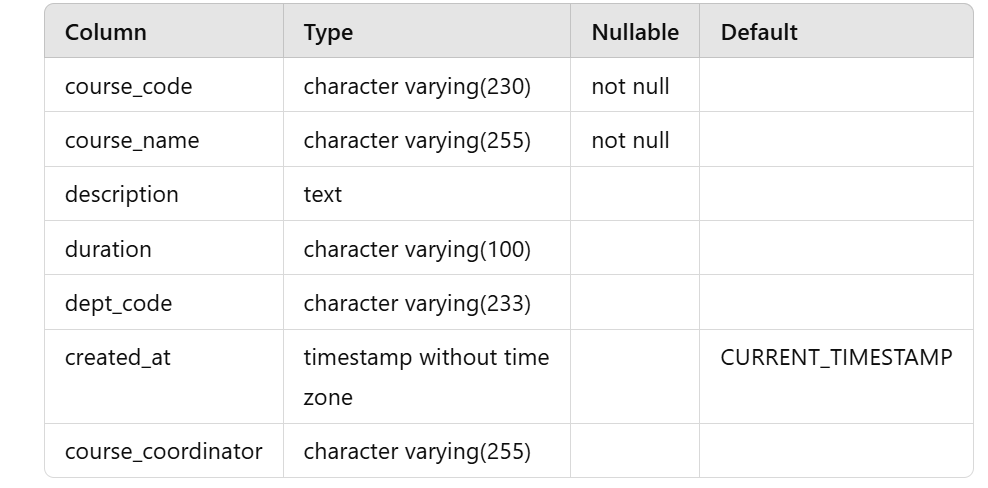
**Final Thoughts:** This project can be considered as a **major project** because it deals with important aspects of college management and addresses real-world issues in educational institutes. It is practical and provides a well-rounded experience in web development, database management, and handling different user roles.

**Database Structure:**

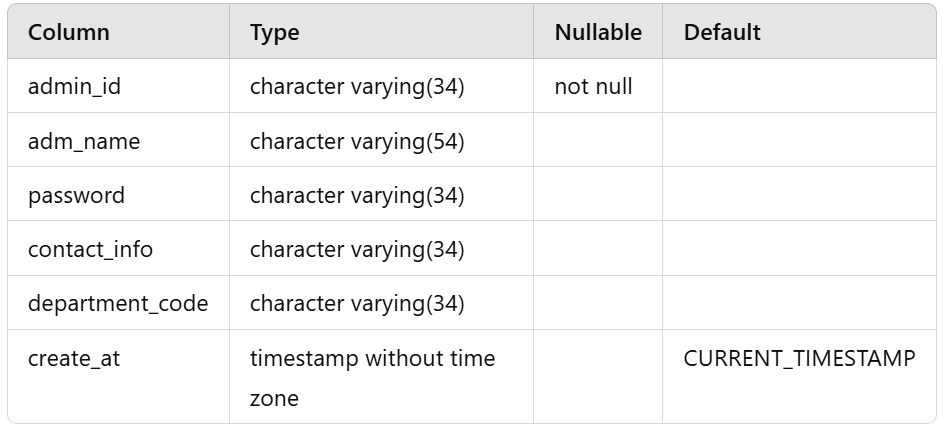
1. **departments**: Stores college departments information.



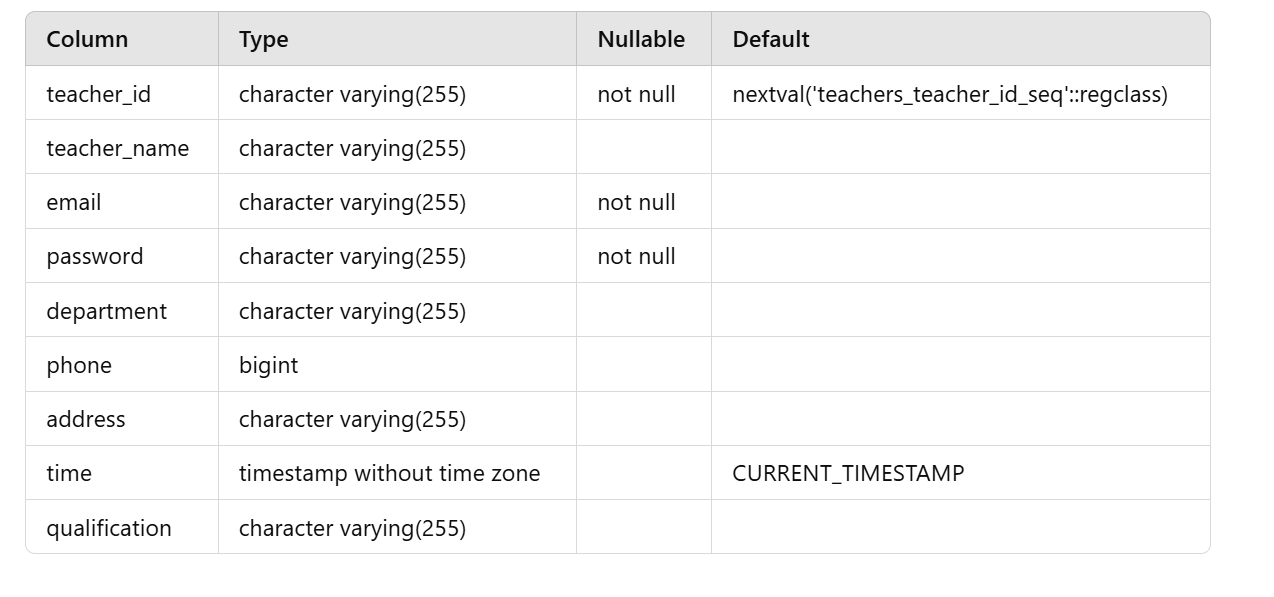
1. **course**: Stores information about available courses.



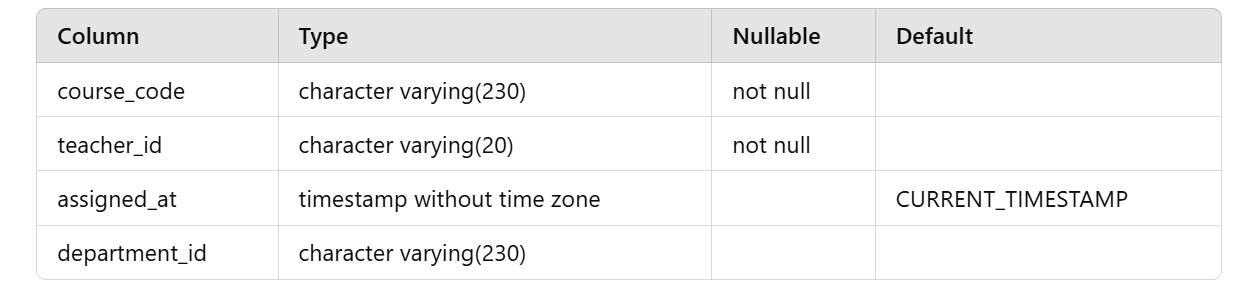
1. **admin**: Stores Admin details.



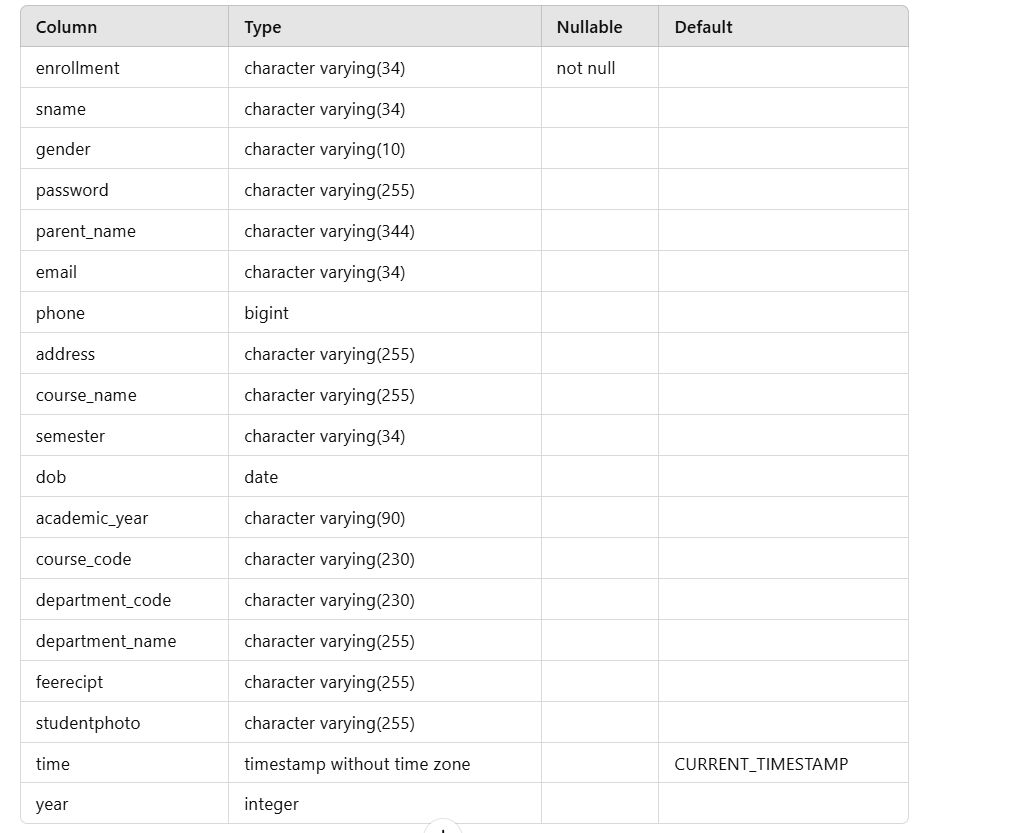
1. **teachers**: Stores teacher information and their assigned courses.



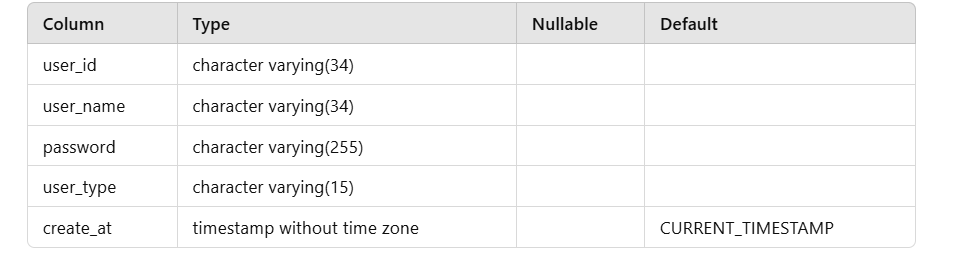
1. **course\_teacher**: Stores the relationship between courses and assigned teachers.



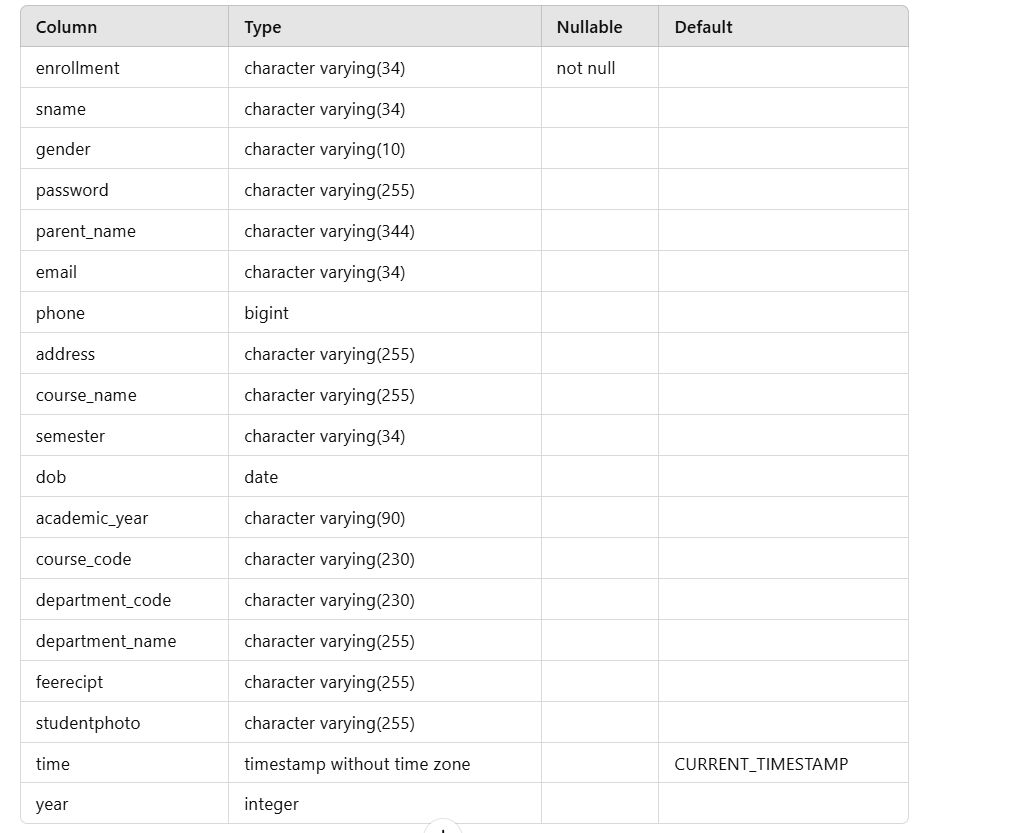
1. **student**: Stores data of verified students.



1. **sy\_user**: Stores user login details (admin, teacher, student).



1. **tempstudent**: Stores unverified student data before admin approval.



 **Department** → **Course** (One-to-Many)

 **Department** → **Admin** (One-to-Many)

 **Department** → **Teacher** (One-to-Many)

 **Department** → **Student** (One-to-Many)

 **Department** → **Tempstudent** (One-to-Many)

 **Course** → **Student** (One-to-Many)

 **Course** ↔ **Teacher** (Many-to-Many via **Course\_Teacher**)

 **Course** → **Classes** (One-to-Many)

 **Teacher** → **Classes** (One-to-Many)

 **Sy\_user** → **Student** (One-to-One)

 **Sy\_user** → **Teacher** (One-to-One)

 **Sy\_user** → **Admin** (One-to-One)