🔐 DATABASE SCHEMA FOR AUTHENTICATION SYSTEM

We'll use the following main tables:

1. Users
2. Roles
3. Roles
4. RefreshTokens
5. PasswordResetTokens
6. UserLogins
7. UserVerifications

1. 📄 Users Table

|  |  |  |  |
| --- | --- | --- | --- |
| Column Name | Data Type | Constraints | Description |
| Id | UNIQUEIDENTIFIER | PK, NOT NULL | Primary key, use NEWID() or GUID |
| Email | VARCHAR(255) | UNIQUE, NOT NULL | User’s unique email address |
| UserName | VARCHAR(100) | NOT NULL | Display name |
| PasswordHash | VARCHAR(512) | NOT NULL | Hashed password (never store plain text!) |
| IsEmailVerified | BIT | DEFAULT 0 | Email verification flag |
| IsPhone Verified | BIT | DEFAULT 0 | Phone verification flag |
| CreatedAt | DATETIME | NOT NULL DEFAULT GETDATE() | Record creation timestamp |
| UpdatedAt | DATETIME | NULL | Optional update timestamp |
| IsActive | BIT | NOT NULL DEFAULT 1 | To support soft delete |
| RoleId | INT | FK → Roles(Id), PK | Linked role |

Either Email or Phone must be provided, but not both necessarily.

2. 🔐 Roles Table

|  |  |  |  |
| --- | --- | --- | --- |
| Column Name | Data Type | Constraints | Description |
| Id | INT | PK, IDENTITY(1,1) | Auto-increment role id |
| Name | VARCHAR(50) | UNIQUE, NOT NULL | Role name (Admin, User, etc.) |

4. 🔁 RefreshTokens Table (for JWT refresh)

|  |  |  |  |
| --- | --- | --- | --- |
| Column Name | Data Type | Constraints | Description |
| Id | UNIQUEIDENTIFIER | PK, NOT NULL | Token Id |
| UserId | UNIQUEIDENTIFIER | FK → Users(Id), NOT NULL | Associated user |
| Token | VARCHAR(512) | NOT NULL | Secure random token |
| ExpiresAt | DATETIME | NOT NULL | Token expiry |
| IsRevoked | BIT | DEFAULT 0 | If token was revoked |
| CreatedAt | DATETIME | DEFAULT GETDATE() | Created timestamp |

5. 🔐 PasswordResetTokens Table

|  |  |  |  |
| --- | --- | --- | --- |
| Column Name | Data Type | Constraints | Description |
| Id | UNIQUEIDENTIFIER | PK, NOT NULL | Token Id |
| UserId | UNIQUEIDENTIFIER | FK → Users(Id), NOT NULL | Linked user |
| Token | VARCHAR(512) | NOT NULL | Secure token for reset |
| ExpiresAt | DATETIME | NOT NULL | Validity time |
| IsUsed | BIT | DEFAULT 0 | Track if token is already used |
| CreatedAt | DATETIME | DEFAULT GETDATE() | Timestamp |

🔑 2. UserLogins Table (Authentication Credentials Table)

|  |  |  |  |
| --- | --- | --- | --- |
| Column Name | Data Type | Constraints | Description |
| Id | UNIQUEIDENTIFIER | PK, NOT NULL | Unique login record |
| UserId | UNIQUEIDENTIFIER | FK → Users(Id), NOT NULL | Linked user |
| LoginType | VARCHAR(50) | NOT NULL | 'Email', 'Phone', 'Google', 'Facebook' |
| ProviderKey | VARCHAR(255) | NOT NULL | Email, Phone, or Third-Party unique key |
| PasswordHash | VARCHAR(512) | NULL (only for email/phone login) | Hashed password for local login only |
| CreatedAt | DATETIME | DEFAULT GETDATE() | When this login was created |
| IsActive | BIT | DEFAULT 1 | Soft delete |

✅ Unified Table: UserVerifications

This table handles:

* 📧 Email verification (via link with token)
* 📱 Phone verification (via OTP)
* 🔄 Can be extended for other purposes like 2FA, password reset, etc.

🧱 Table: UserVerifications

|  |  |  |  |
| --- | --- | --- | --- |
| Column Name | Type | Constraints | Description |
| Id | UNIQUEIDENTIFIER | PK | Primary key |
| UserId | UNIQUEIDENTIFIER | FK → Users(Id) | Linked to user |
| ContactType | VARCHAR(10) | NOT NULL | 'Email' or 'Phone' |
| ContactValue | VARCHAR(255) | NOT NULL | Email address or phone number |
| VerificationCode | VARCHAR(100) | NOT NULL | Token (for email) or OTP (for phone) |
| CodeType | VARCHAR(20) | NOT NULL | 'Token' or 'OTP' |
| Purpose | VARCHAR(30) | NOT NULL | 'AccountVerification', 'PasswordReset' |
| ExpiresAt | DATETIME | NOT NULL | Expiry time |
| IsUsed | BIT | DEFAULT 0 | Whether the code has been used |
| CreatedAt | DATETIME | DEFAULT GETDATE() | Timestamp |