Authentication System & Database Design

Multi■Vendor E■Commerce Platform (Production■Ready Architecture)

Final Auth & Role Strategy

• No wallet balance system

• /register → USER only

• /seller/register → SELLER only

• ADMIN & QA_ADMIN → hidden / internal registration

• SUPER_ADMIN → created manually via database only

Roles Overview

USER → Buyer / Customer

SELLER → Vendor

ADMIN → Ops / Support / Content

QA_ADMIN → Testing / Staging

SUPER_ADMIN → System Owner (DB only)

Registration Rules

USER → Public API (/register)

SELLER → Public API (/seller/register)

ADMIN → Internal API only

QA_ADMIN → Internal API only

SUPER_ADMIN → Database insertion only

User Authentication APIs

```
POST /v1/auth/register
POST /v1/auth/login
POST /v1/auth/logout
POST /v1/auth/refresh
POST /v1/auth/send-otp
POST /v1/auth/verify-otp
POST /v1/auth/request-password-reset
POST /v1/auth/reset-password
POST /v1/auth/change-password
GET /v1/auth/sessions
DELETE /v1/auth/sessions/{sessionId}
```

Seller Authentication APIs

```
POST /v1/seller/register
POST /v1/seller/kyc
GET /v1/seller/kyc/status
```

Admin & QA Admin APIs (Hidden)

```
POST /v1/internal/admin/create
POST /v1/internal/qa-admin/create
PUT /v1/internal/users/{id}/disable
PUT /v1/internal/users/{id}/enable
```

Database Structure (Postgres / Neon)

users (Master Table)

```
id UUID PK
email UNIQUE
phone UNIQUE
password_hash
role ENUM
status ENUM
is_email_verified
is_phone_verified
two_fa_enabled
last_login_at
created_at
updated_at
```

user_profiles

```
user_id PK/FK
full_name
profile_image
gender
dob
created_at
```

user_addresses

```
id PK
user_id FK
address fields...
```

seller_profiles

```
user_id PK/FK
store_name
store_slug
business_type
gst_number
pan_number
kyc_status
rating
created_at
```

seller_bank_accounts

```
id PK
seller_id FK
bank_name
account_number
ifsc_code
is_primary
```

admin_profiles

```
user_id PK/FK
full_name
department
created_at
```

qa_admin_profiles

```
user_id PK/FK
environment
access_level
created_at
```

super_admin_profiles

```
user_id PK/FK
owner_level
created_at
```

Security & Audit Tables

```
login_sessions
audit_logs
```

JWT Payload

```
{
userId,
role,
status
}
```

Middleware

• authGuard

• roleGuard(['ADMIN','SUPER_ADMIN'])

Final Confirmation Checklist

✔ Separate register APIs

✔ Clean DB normalization

✔ Neon compatible

✔ Secure admin isolation

✔ Easy future scaling