

## ✓ Was ich bereits gemacht habe:

### 🔧 1. Django-Projekt Setup:

- Django + Django REST Framework installiert ✓
- Projekt und App core erstellt ✓
- AUTH\_USER\_MODEL angepasst und konfiguriert ✓
- settings.py korrekt eingerichtet ✓
- Lokaler Server gestartet ✓

### 📦 2. Datenmodelle (models.py):

- Benutzerdefiniertes User-Modell ✓
- ServiceType-Modell ✓
- ProviderProfile-Modell ✓
- Request-Modell ✓

### 🔗 3. Migrationen:

- Migrationen erstellt und angewendet ✓

### 🔒 4. Authentifizierung:

- Superuser erstellt und in der Shell getestet ✓

### 🔧 5. Lokale Tests:

- Server ausgeführt, Admin-Bereich aufgerufen ✓

### 🌐 6. API:

- RequestViewSet erstellt ✓
- Router in core/urls.py konfiguriert ✓
- In backend/urls.py eingebunden ✓

---

## 📅 Was noch fehlt:

### 💎 7. API komplettieren:

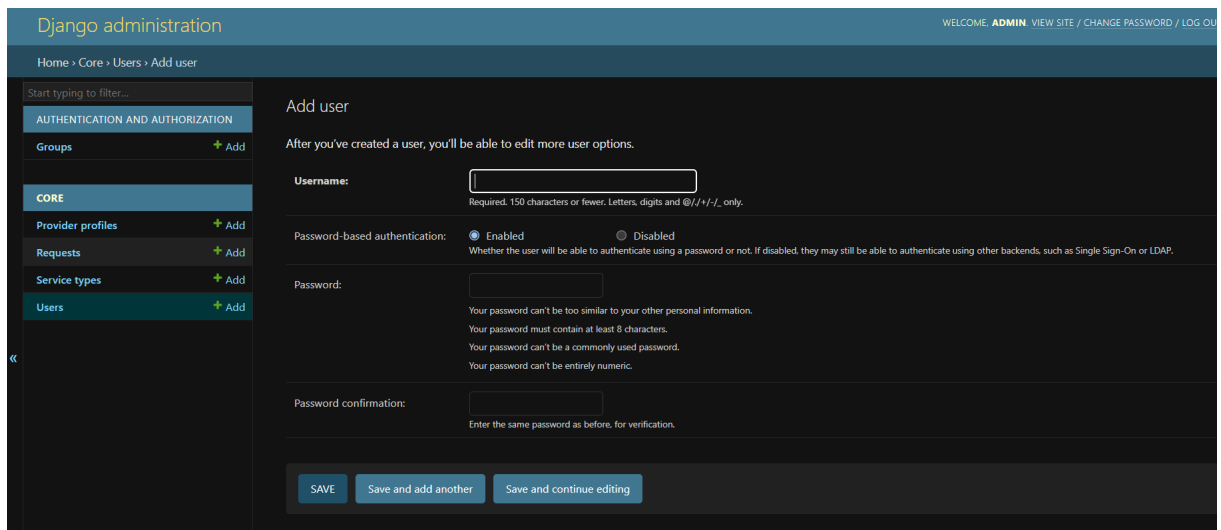
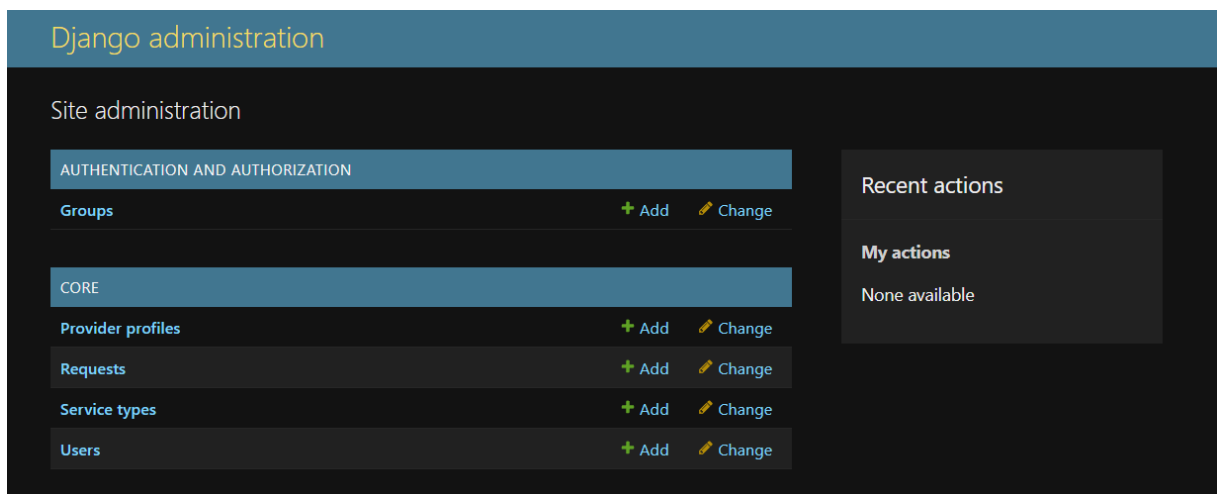
- ViewSets für User, ServiceType, ProviderProfile
- Serializers für alle Modelle
- Authentifizierung mit Token oder JWT
- Individuelle Validierungen (Region, Status etc.)

## ◆ 8. Erweiterte Funktionen:

- Datei-Uploads (optional)
- Business-Logik (z. B. automatisches Matching)
- Filter (z. B. nach Region oder Status)
- Berechtigungen: Wer darf was sehen oder senden

## ◆ 9. Abschluss:

- API-Dokumentation mit Swagger/OpenAPI
- (Optional) Automatisierte Tests



[Api Root](#) / [Request List](#)

## Request List

OPTIONS

GET ▾

GET /api/requests/

```
HTTP 200 OK
Allow: GET, POST, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept

[]
```

Raw data

HTML form

Service type

Company name

Address

Contact person

Phone number

Email

Status

Gesendet ▾

Client

admin ▾

POST

[Api Root](#) / [Request List](#)

## Request List

OPTIONS

GET ▾

GET /api/requests/

```
HTTP 200 OK
Allow: GET, POST, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept

[]
```

Raw data

HTML form

Service type

Company name

Address

Contact person

Phone number

Email

Status

Gesendet ▾

Client

admin ▾

POST

