

---

# medrec (Django backend)

This is a starter Django backend for your graduation project: users upload an image of a medicine and receive an AI response (name, info, etc.). The repository contains authentication (JWT), image upload API, and a placeholder AI inference service.

## Setup Complete

The backend is already set up and running at **http://127.0.0.1:8000/**

### Superuser credentials:

- Username: admin
- Password: admin123

If you need to set up again:

```
POWERSHELL
python -m venv venv
venv\Scripts\Activate
pip install -r requirements.txt
cd medrec
python manage.py migrate
python manage.py runserver
```

## API Endpoints

### Authentication:

- POST /auth/register/ - Register new user
  - Body: `{"username": "user1", "email": "user@example.com", "password": "pass123"}`
- POST /auth/token/ - Get JWT access token
  - Body: `{"username": "user1", "password": "pass123"}`

- 
- Returns: `{"access": "...", "refresh": "..."}"`
  - POST `/auth/token/refresh/` - Refresh access token
  - Body: `{"refresh": "your-refresh-token"}`
  - GET `/auth/me/` - Get current user profile (requires authentication)
  - Header: `Authorization: Bearer`

### Medicine Image Upload:

- POST `/api/uploads/new/` - Upload medicine image (requires authentication)
- Header: `Authorization: Bearer`
- Body: multipart form with `image` field
- Returns: Upload record with AI inference `result`
- GET `/api/uploads/` - List your uploads (requires authentication)
- Header: `Authorization: Bearer`

### Admin Panel:

- `http://127.0.0.1:8000/admin/` - Django admin (login with superuser credentials)

## Testing the API

You can test with **Postman**, **curl**, or any HTTP client:

### 1. Register a new user

#### POWERSHELL

```
curl -X POST http://127.0.0.1:8000/auth/register/ `
  -H "Content-Type: application/json" `
  -d '{"username":"testuser","email":"test@example.com","password":"testpass123"}
```

### 2. Get access token

#### POWERSHELL

```
curl -X POST http://127.0.0.1:8000/auth/token/ `
  -H "Content-Type: application/json" `
```

---

```
-d '{"username\":\"testuser\",\"password\":\"testpass123\"}'
```

Save the `access` token from the response.

### 3. Upload a medicine image

**POWERSHELL**

```
curl -X POST http://127.0.0.1:8000/api/uploads/new/ `
-H "Authorization: Bearer YOUR_ACCESS_TOKEN" `
-F "image=@path/to/medicine.jpg"
```

The response will include the AI inference result in the `result` field.

### 4. List your uploads

**POWERSHELL**

```
curl http://127.0.0.1:8000/api/uploads/ `
-H "Authorization: Bearer YOUR_ACCESS_TOKEN"
```

## Notes

- Replace `core/ai_service.py::infer` with an actual model or external API call to perform inference.
- Media files are served in development when `DEBUG=True`.