BLOG PLATFORM BACKEND — FULL FEATURE SET WITH ALL DJANGO TOPICS

User & Authentication System

Features

- Custom User Model (email login)
- Roles: Admin, normal_user
- Register, Login, Logout
- Token authentication (your custom TokenRequiredDecorator)
- TokenRequiredMixin + RoleRequiredMixin
- Profile update (name, bio, profile picture)
- Upload profile picture to Cloudinary
- Password reset
- Welcome mail

Blog Post Management

Features

- Create post
- Edit post
- Delete post
- Post status → draft / published / archived
- Category system

Draft/Publish Workflow

- Users creates and drafts and publishes
- Admin can delete a post

Comments System

Features

- Only authenticated users can comment
- Edit & delete your own comments
- Moderation → Admin can block inappropriate comments
- Notifications to post owner when new comment added

Likes System

Features

- Like/unlike a blog post
- Like count display
- User can like only once

Search, Filter & Pagination (Real World)

Features

- Search posts: title, tags, category, author, full text
  `?search=django`

- Filter:

    - by category
    - by tags
    - by author
    - published only

- ○ date range

- Pagination: page=, page_size=

Concepts Included

- DRF SearchFilter
- Filtering backend

Middlewares: Add **3 real-world middlewares**:

1. Request Logging Middleware

- Logs: URL, method, user, IP address

2. Execution Time Middleware

- Add header: X-Execution-Time: 150ms

3. Block Suspicious User Middleware

- Block requests with missing token
- Allow whitelisted URLs (login/register)

8️⃣Custom Decorators & Mixins

Mixins

- AdminRequiredMixin
- OwnerOnlyMixin (for post author only edit/delete)

File Handling + Cloud Storage

Files you'll handle:

- User profile images

Implement:

- File validation (max size)
- Supported types
- Cloudinary configuration
- Auto-delete images when post is deleted (signal)

Notifications System

For:

- New comment on your post
- New follower
- Email notifications

Followers System

Features:

- Feed of posts from followed authors

API Documentation

#Deploy the project