

Post Anything / Issue Management System— Project Specification

1. PROJECT OVERVIEW

The **Post Management System** allows users to submit posts about various concerns, requests, or information.

- Issue – Office furniture is broken
 - Complaint – Unhygienic canteen food, bad smell
 - Post – Lost and Found, Need Blood
- Note: These are just examples; users can post anything.

Admins review and approve posts before they become visible.

Tech Stack:

- Backend: Spring Boot 3 / Spring 6
- Frontend: Angular
- Database: H2 / Any Preferable
- Security: Spring Security 6 (Basic Authentication, optional JWT)
- ORM: Spring Data JPA

2. FUNCTIONAL REQUIREMENTS

- Users can register and log in.
- Users can create posts of different types (lost item, help request, announcement).
- Admins can view all posts, approve or reject them, and assign updates if needed.
- Each post has a status lifecycle: **DRAFT → PENDING_APPROVAL → APPROVED → REJECTED → CLOSED.**
- Role-based access control (USER vs ADMIN).

3. SYSTEM DESIGN & ARCHITECTURE

Layers:

- **Controller:** REST endpoints for Posts, Comments, and Authentication.
- **Service:** Business logic, validation, and workflow handling.
- **Repository:** JPA repositories for persistence.
- **Security:** Basic Auth with optional JWT.

4. DATABASE SCHEMA

5. API SPECIFICATIONS

6. POST WORKFLOW

DRAFT → PENDING_APPROVAL → APPROVED → RESOLVED
↓
REJECTED

Roles & Permissions:

Action	USER	ADMIN
Create Post	✓	✓

Submit Post for Approval	<input checked="" type="checkbox"/>	-
Approve / Reject Post	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Assign / Update Post	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Add Comment	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
View Post	<input checked="" type="checkbox"/> (own + all approved)	<input checked="" type="checkbox"/> (all)

7. SECURITY FLOW

- **Basic Authentication:** default; credentials validated via DB .
- **JWT (Optional):** replace Basic Auth with token-based authentication.
- **Access Control:** /auth/** public, /api/** protected.

8. MILESTONES / EVALUATION

Week 1: Setup Spring Boot project, H2 DB or any preferable, Entities, CRUD operations.

Week 1: Add Security (Basic Auth), Role-based access.

Week 2: Add Comments, filtering, workflow, and exception handling.

Week 2 (Optional): JWT authentication, notifications, search filters.