E-Gov Grievance Redressal System

A Scalable, Reactive Microservices Approach to Transparent Governance

Presented By:

Yati

CHUBB

Java FSD

Amity University

Problem Statement

Current Challenges:

- Lack of transparency in grievance tracking.
- Delayed responses and SLA breaches without accountability.
- Manual and fragmented processes.
- Poor communication between citizens and government departments.

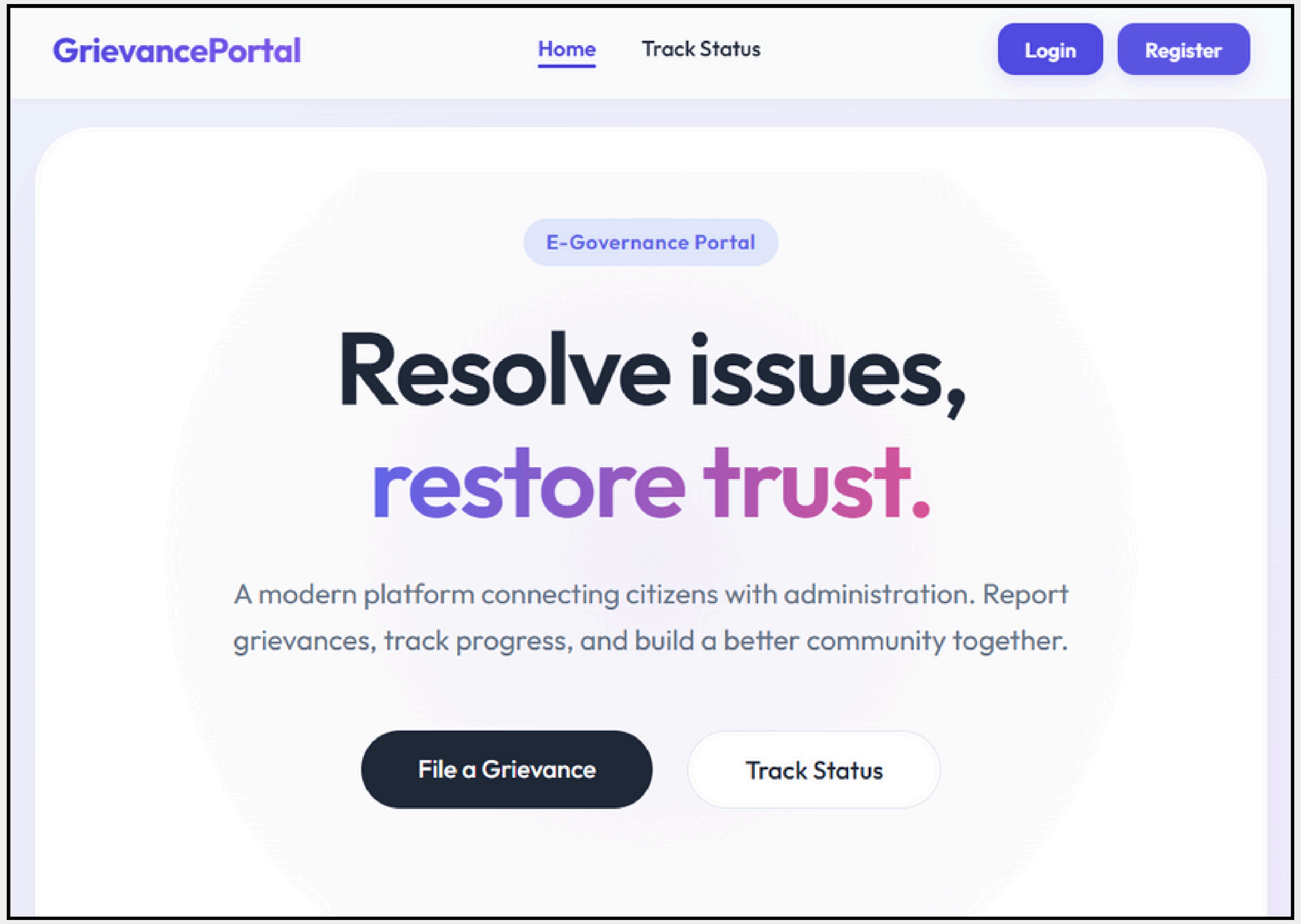
Need: A centralized, automated, and real-time system to handle public concerns efficiently.

Solution Overview

Project Goal: To provide a robust platform for citizens to file, track, and resolve grievances with escalation feature and real-time notifications.

Key Pillars:

- Transparency: Real-time tracking for citizens.
- Accountability: Role-based access and SLA monitoring.
- Scale: Microservices architecture for high availability.
- Engagement: Automated email notifications.



Resolve issues, restore trust.

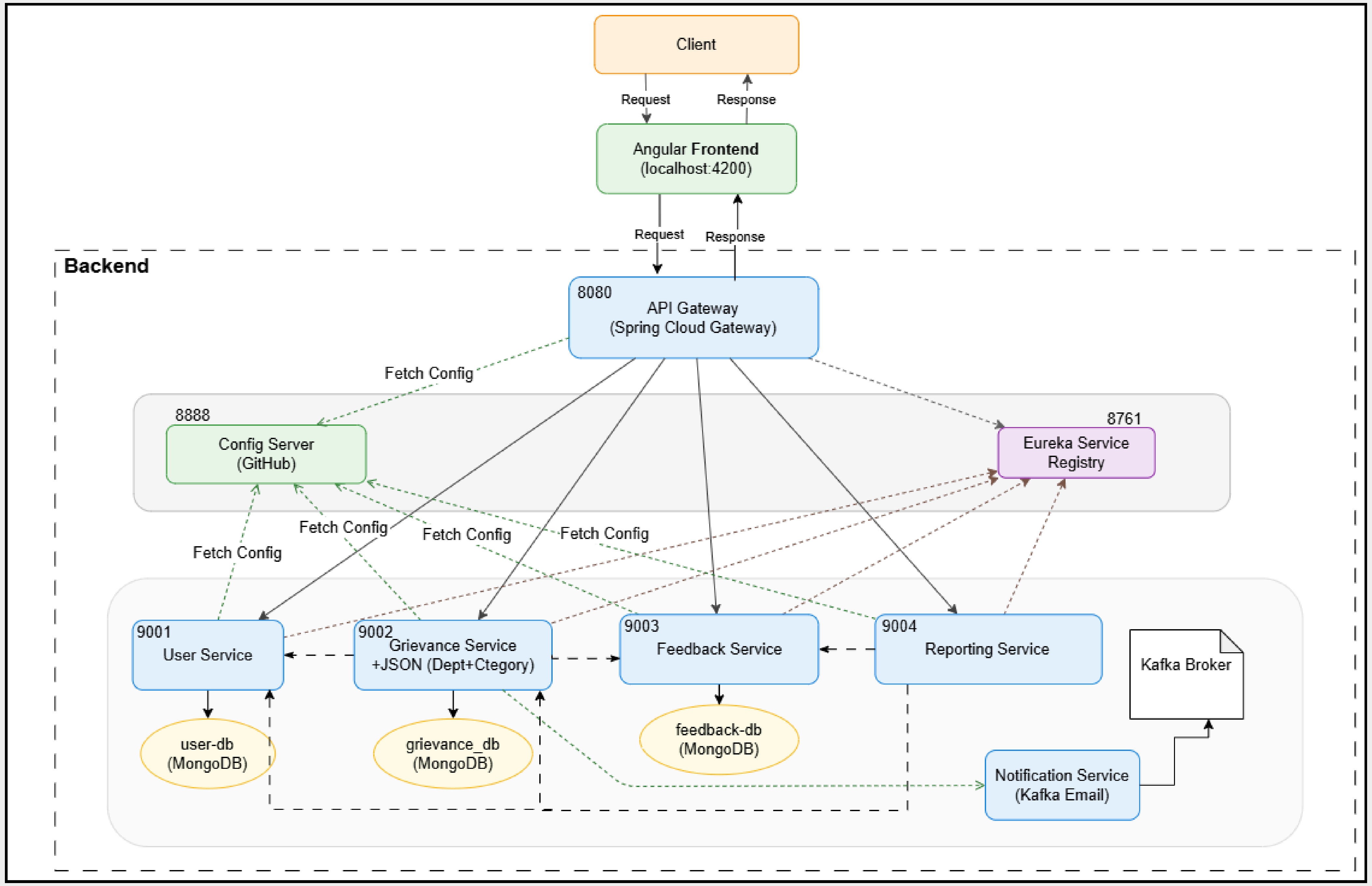
A modern platform connecting citizens with administration. Report grievances, track progress, and build a better community together.

Technology Stack

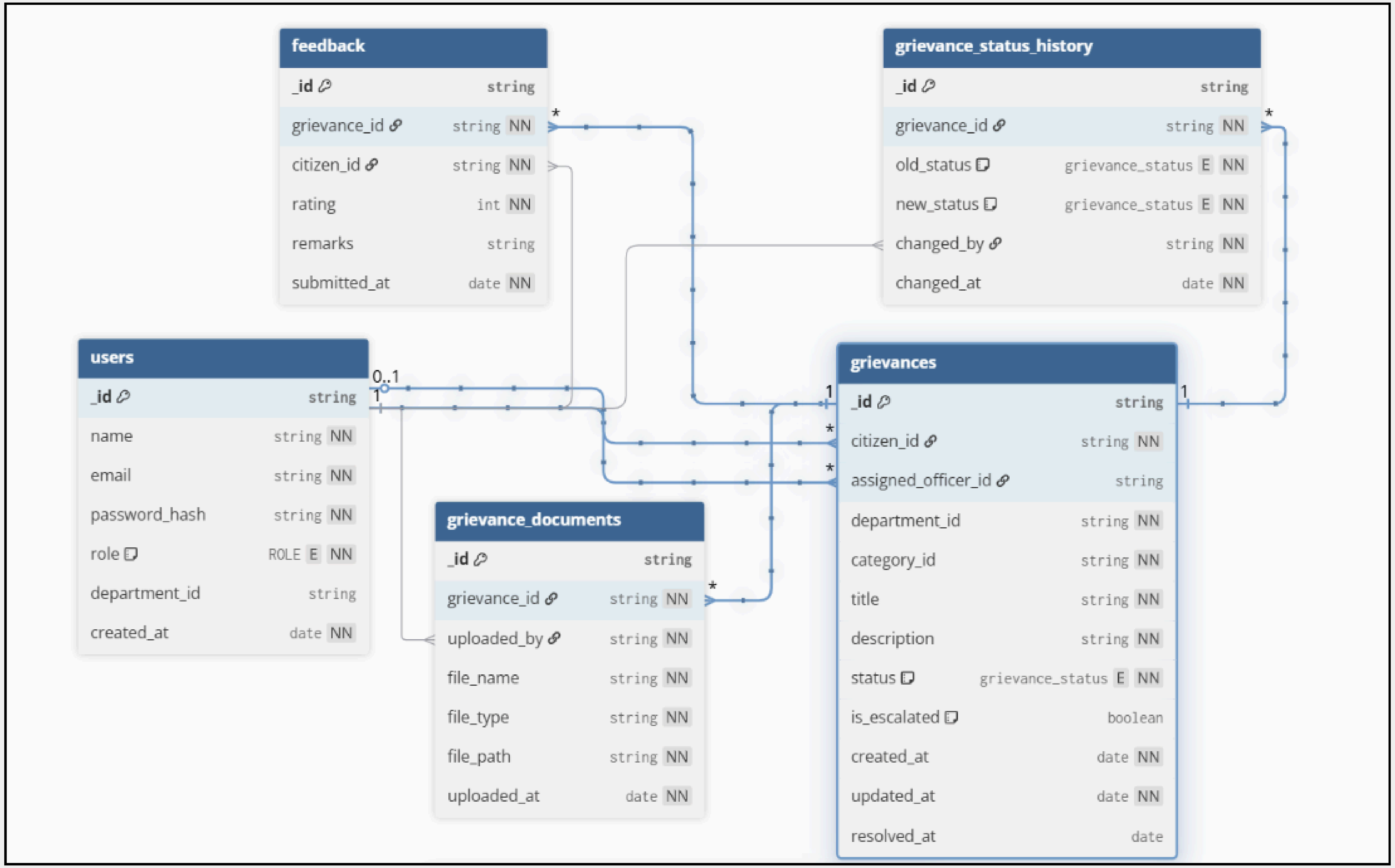
- **Backend**: Java 21, Spring Boot 3.4, Spring Cloud (Gateway, Eureka, Config Server).
- **Reactive Programming:** Project Reactor (Mono/Flux), Spring WebFlux.
- **Frontend**: Angular 21
- **Database**: MongoDB 6.0 (NoSQL for flexible schema).
- **Messaging**: Apache Kafka (Event-driven status updates).
- **Infrastructure**: Docker, Docker Compose, Jenkins.

Microservices Architecture

- **API Gateway**: Central entry point, JWT handling, routing.
- **User Service:** Identity Management, RBAC
- **Grievance Service**: Core business logic, status management, file uploads.
- **Notification Service:** Kafka-driven
- **Reporting Service**: Analytics/summaries.
- **Eureka**: Service discovery.



Database Design



feedback		
_id	string	
grievance_id	string	NN
citizen_id	string	NN
rating	int	NN
remarks	string	
submitted_at	date	NN

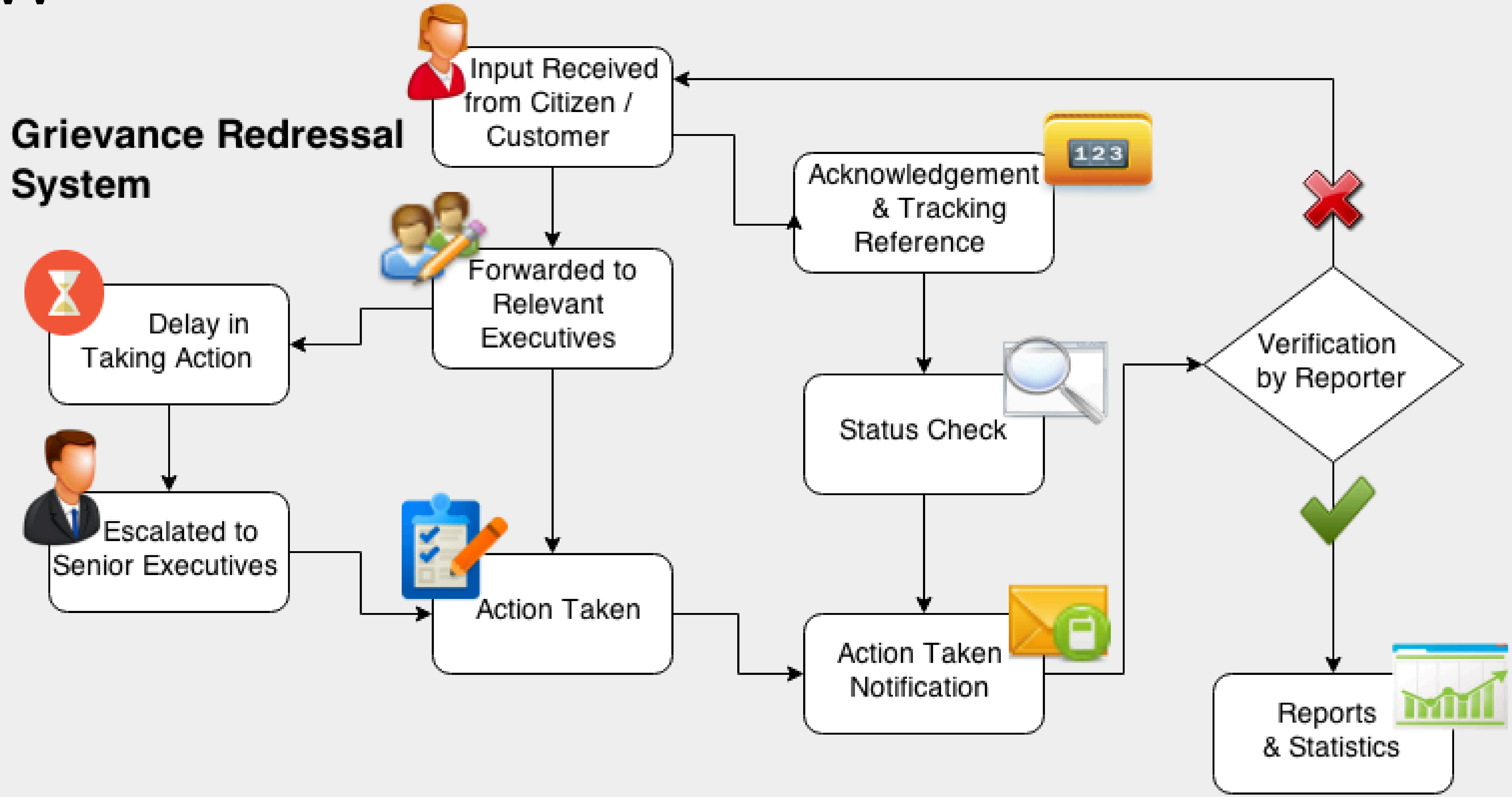
grievance_status_history		
_id	string	
grievance_id	string	NN
old_status	grievance_status E	NN
new_status	grievance_status E	NN
changed_by	string	NN
changed_at	date	NN

users		
_id	string	
name	string	NN
email	string	NN
password_hash	string	NN
role	ROLE E	NN
department_id	string	
created_at	date	NN

grievance_documents		
_id	string	
grievance_id	string	NN
uploaded_by	string	NN
file_name	string	NN
file_type	string	NN
file_path	string	NN
uploaded_at	date	NN

grievances		
_id	string	
citizen_id	string	NN
assigned_officer_id	string	
department_id	string	NN
category_id	string	NN
title	string	NN
description	string	NN
status	grievance_status E	NN
is_escalated	boolean	
created_at	date	NN
updated_at	date	NN
resolved_at	date	

Flow


Grievance Redressal System
Input Received from Citizen / Customer
Acknowledgement & Tracking Reference
123
Forwarded to Relevant Executives
Delay in Taking Action
Escalated to Senior Executives
Action Taken
Status Check
Action Taken Notification
Verification by Reporter
Reports & Statistics

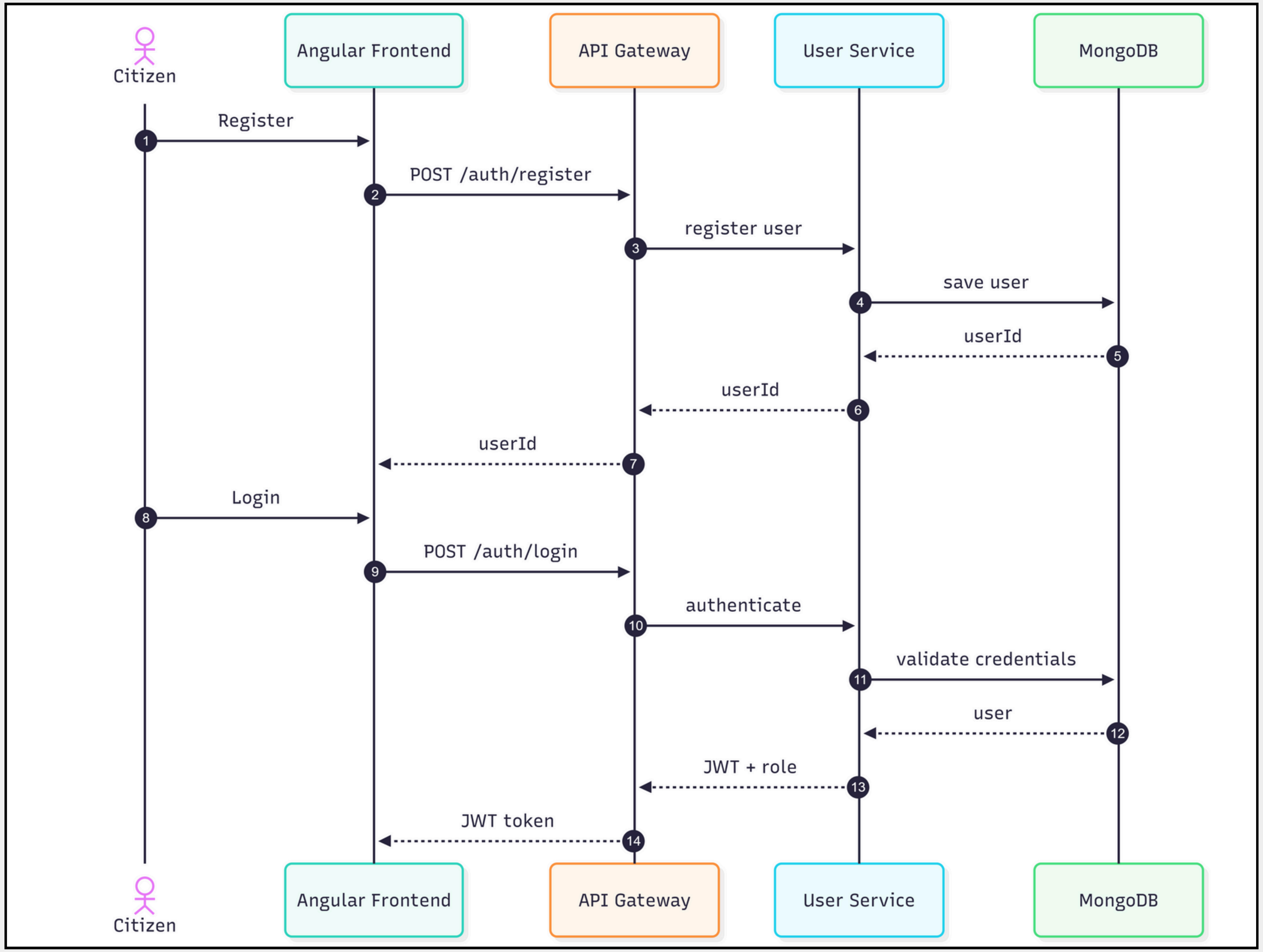
Roles & Responsibilities

Role	Responsibilities
Citizen	Register/Login , Lodge grievances, upload documents, view status, escalate delays, close/reopen grievances, give feedback
Officer	Review, process, and resolve grievances
Supervisor	Assign grievances, monitor escalations, view reports
Admin	Manage users and roles, view reports

Once logged in, everything the user can see or do depends on their role

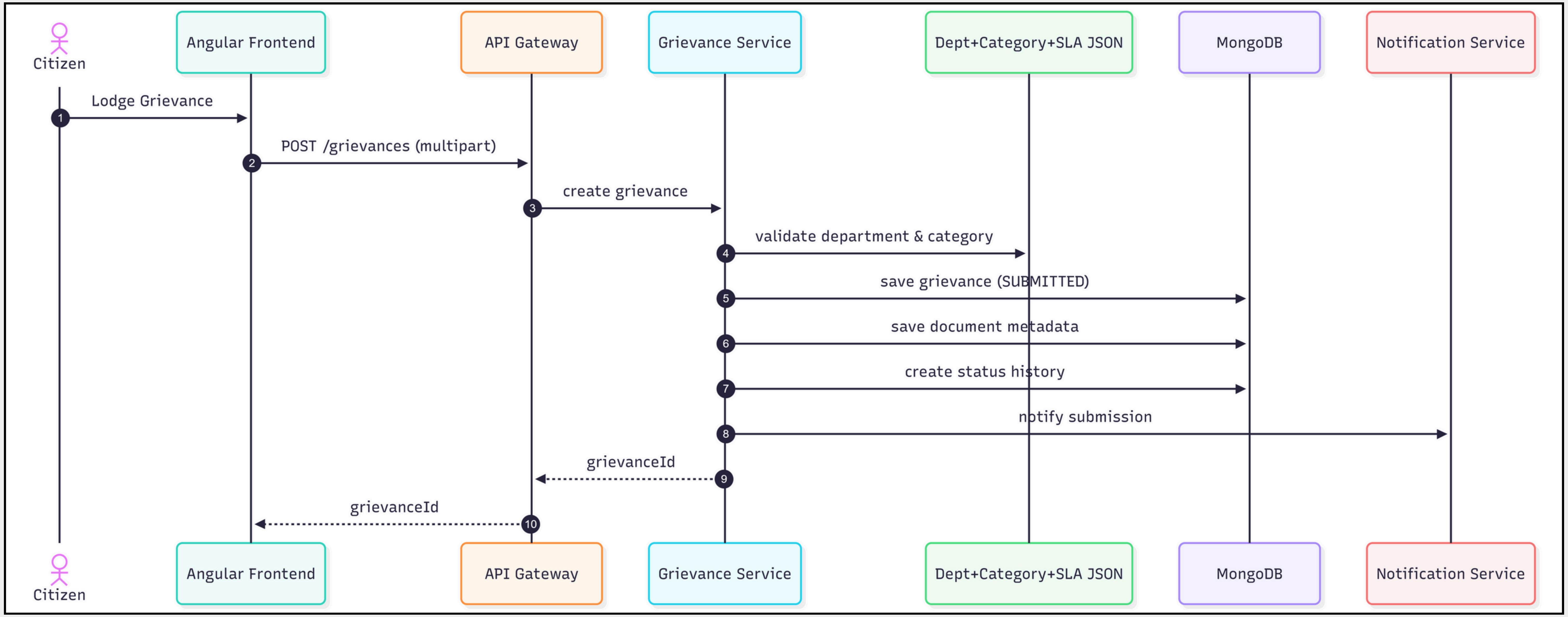
User Registration & Login

- Register → user enters details → password encrypted → ROLE = CITIZEN
- Admin create users → Officer / Supervisor / Admin
- Login → credentials verified → JWT (userId + role)
- Access → JWT sent with requests → role checked → allow/deny



Grievance Creation

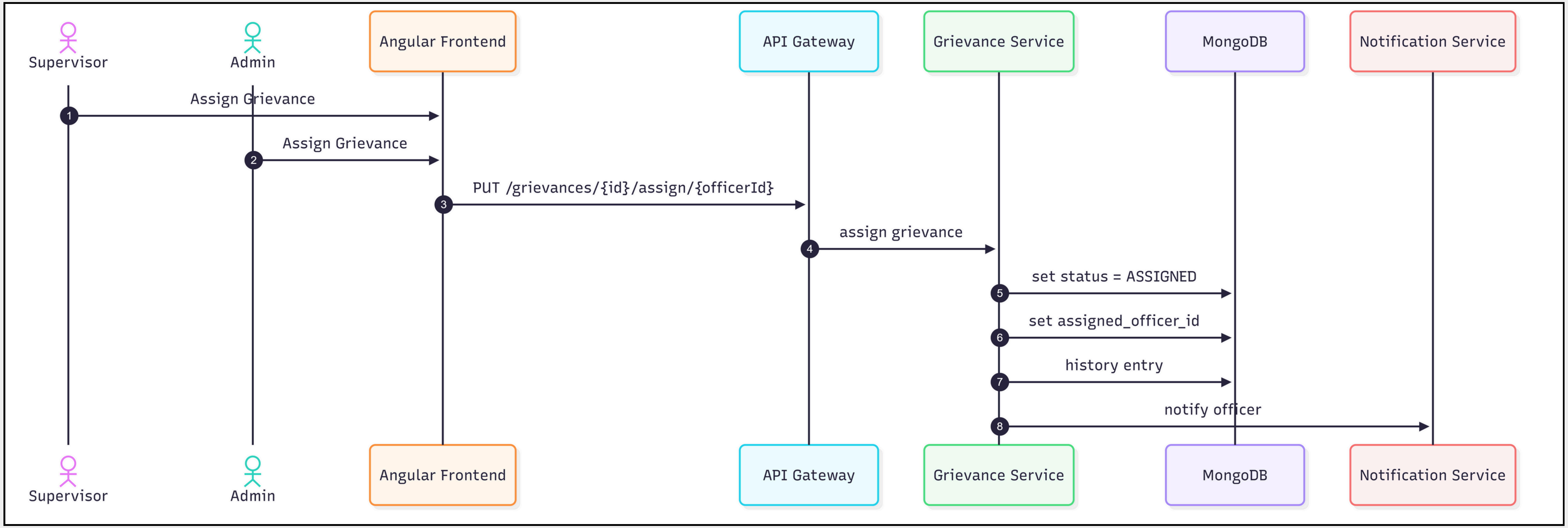
Citizen → Lodge Grievance → Enters title, description, department, category → Uploads documents



System → validate → generates id → sets status SUBMITTED → stores document metadata → creates status history → sends email notification

Grievance Assignment

Supervisor/Admin → views SUBMITTED grievances → Assigns grievance → Officer



System → validates officer belongs to department → updates status → ASSIGNED → saves→ assigned_officer_id → logs status history → email

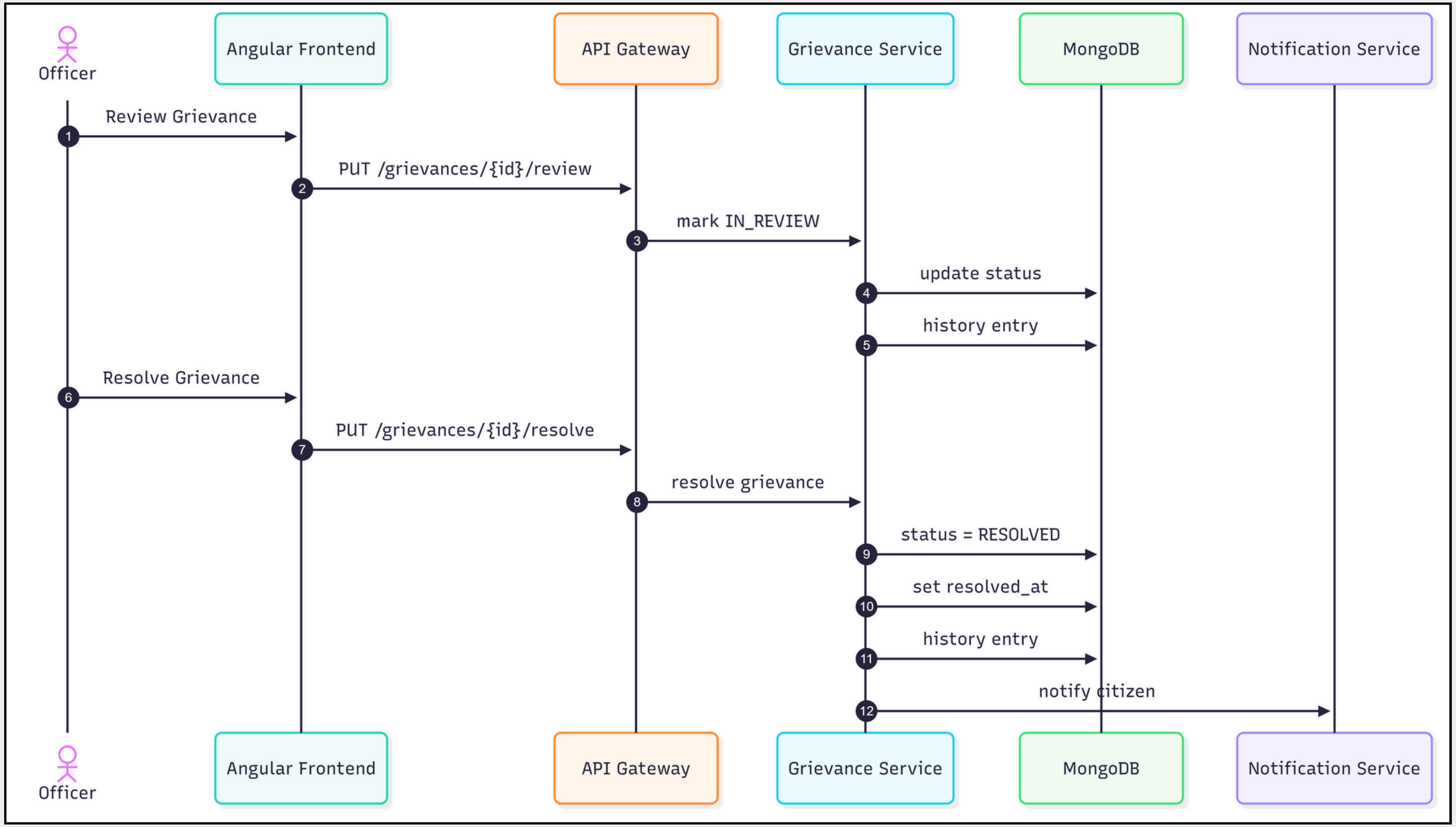
Grievance Review & Resolution

Officer → logs in → views assigned grievances

- Reviews details & documents
- Updates status → IN_REVIEW
- Resolves grievance
- sets status RESOLVED

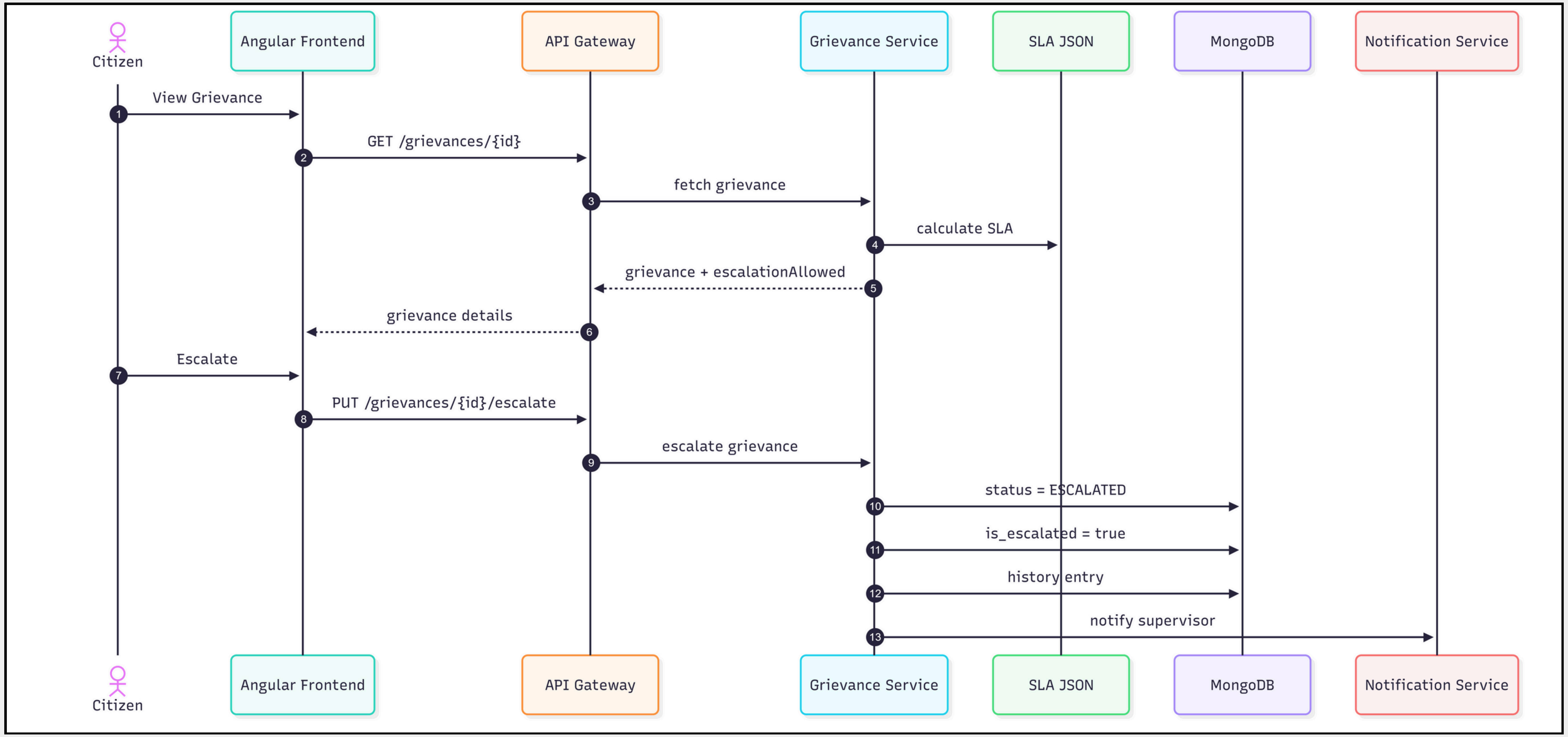
System → saves resolved_at

- logs history
- notifies citizen



SLA Evaluation & Escalation

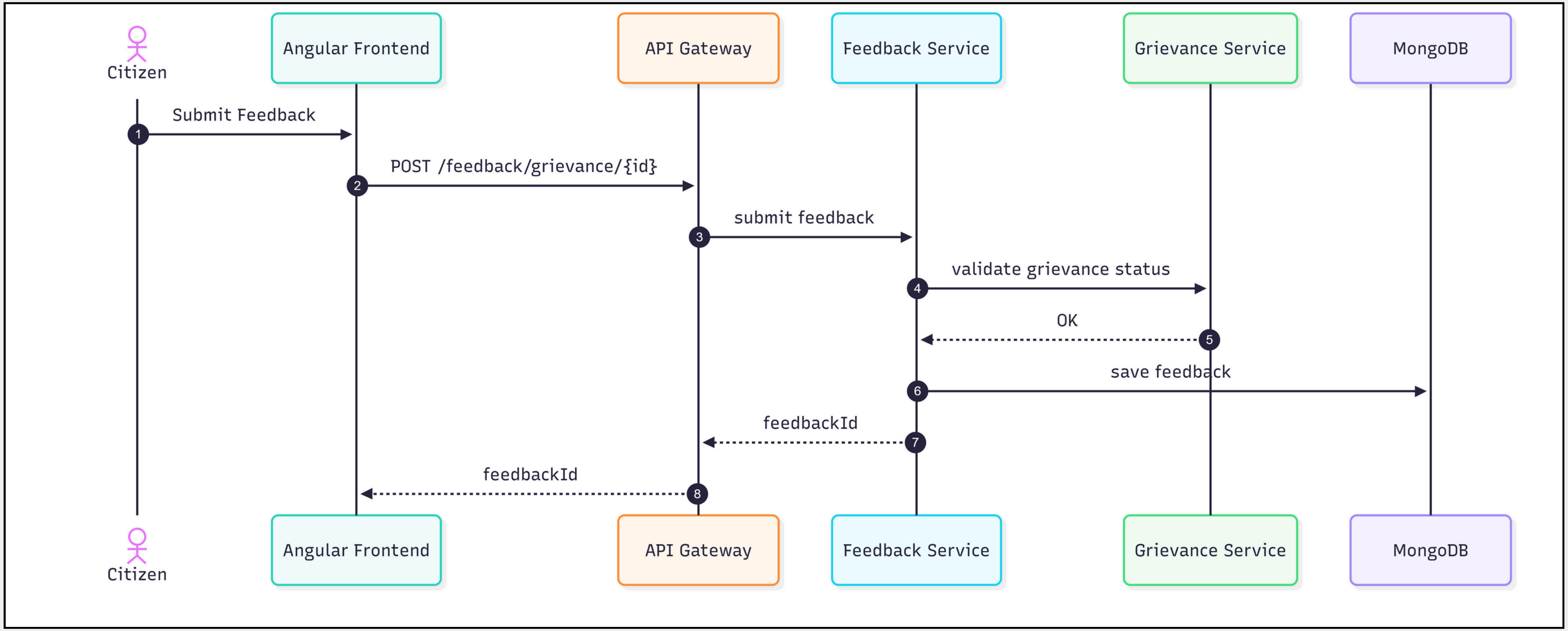
- SLA → defined per category (JSON)
- Escalation (Allowed if: not RESOLVED/CLOSED, SLA exceeded, not already escalated)



On escalate → status ESCALATED → is_escalated = true → history logged → supervisors notified

Feedback Submission

- Citizen → submits feedback after RESOLVED/CLOSED
- Includes rating (1–5) + optional remarks



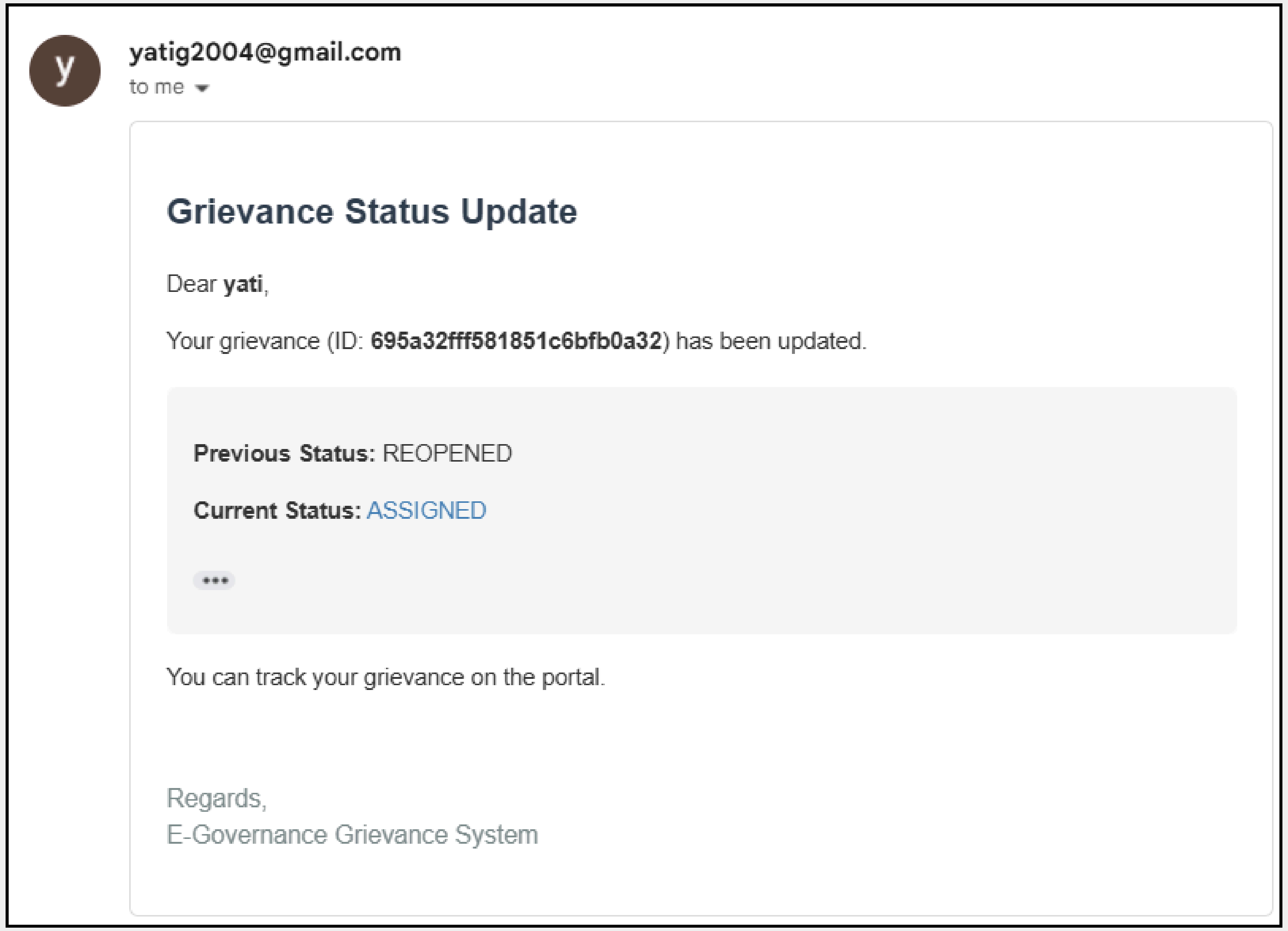
Notifications

Notifications are sent for:

- grievance submission,
- assignment,
- in review,
- resolution,
- escalation,
- reopened

The Notification Service:

- is stateless,
- is triggered only by internal services,
- does not block business flow on failure.



yatig2004@gmail.com
to me

Grievance Status Update

Dear **yati**,

Your grievance (ID: **695a32fff581851c6bfb0a32**) has been updated.

Previous Status: REOPENED

Current Status: ASSIGNED

You can track your grievance on the portal.

Regards,
E-Governance Grievance System

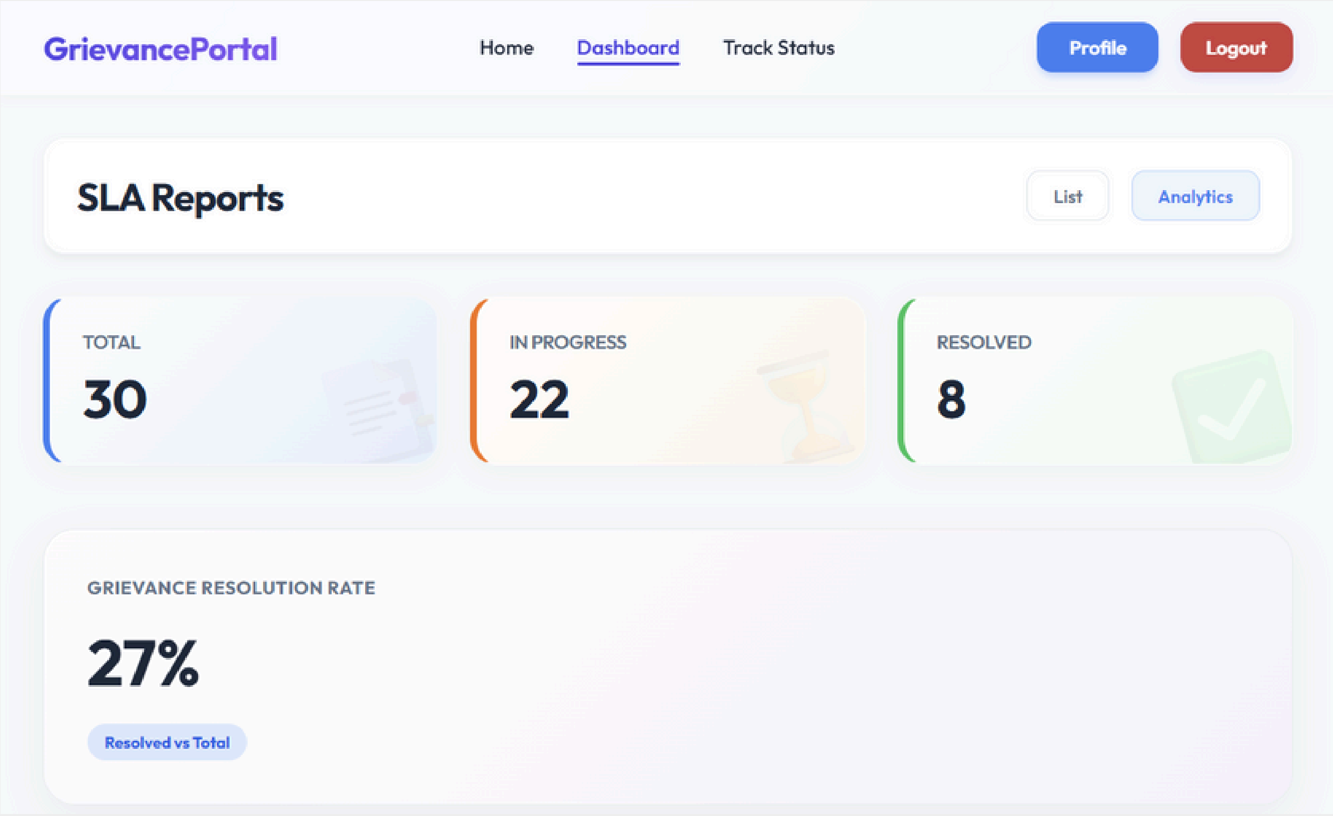
Reporting & Analytics

Users can access dashboards showing:

- grievance counts by status,
- department-wise performance,
- average resolution time, etc

Reporting service:

- is strictly read-only,
- aggregates data from other services.



SLA Reports

TOTAL 30

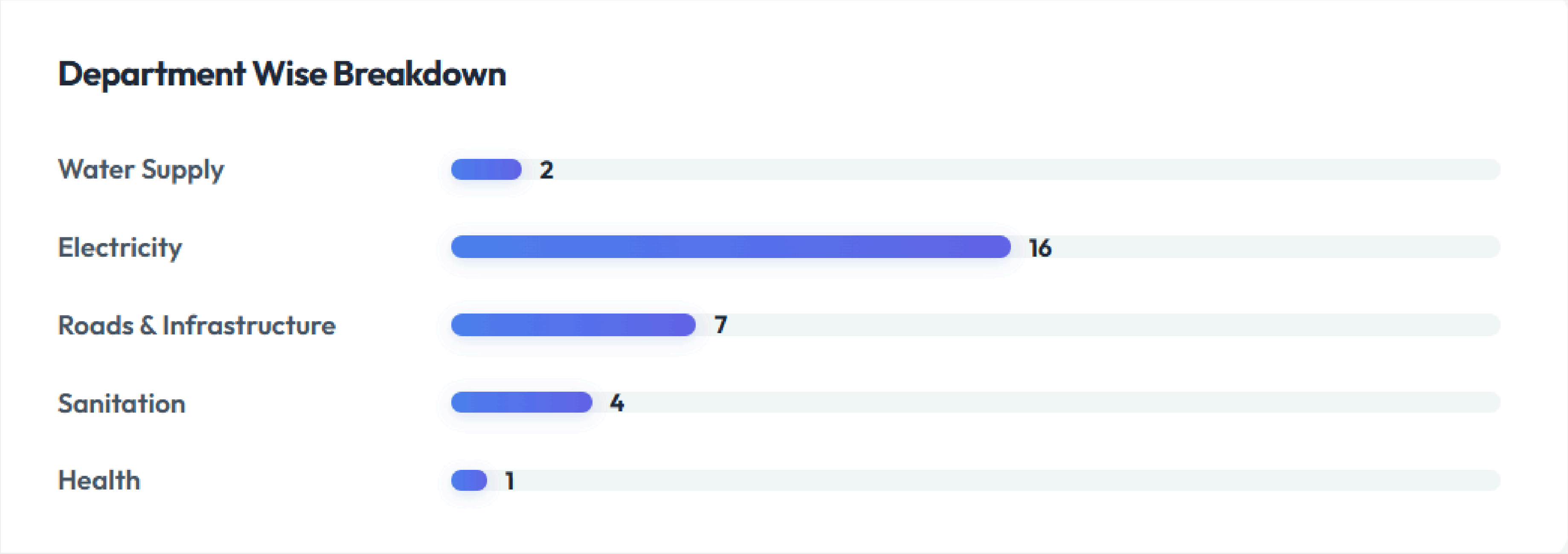
IN PROGRESS 22

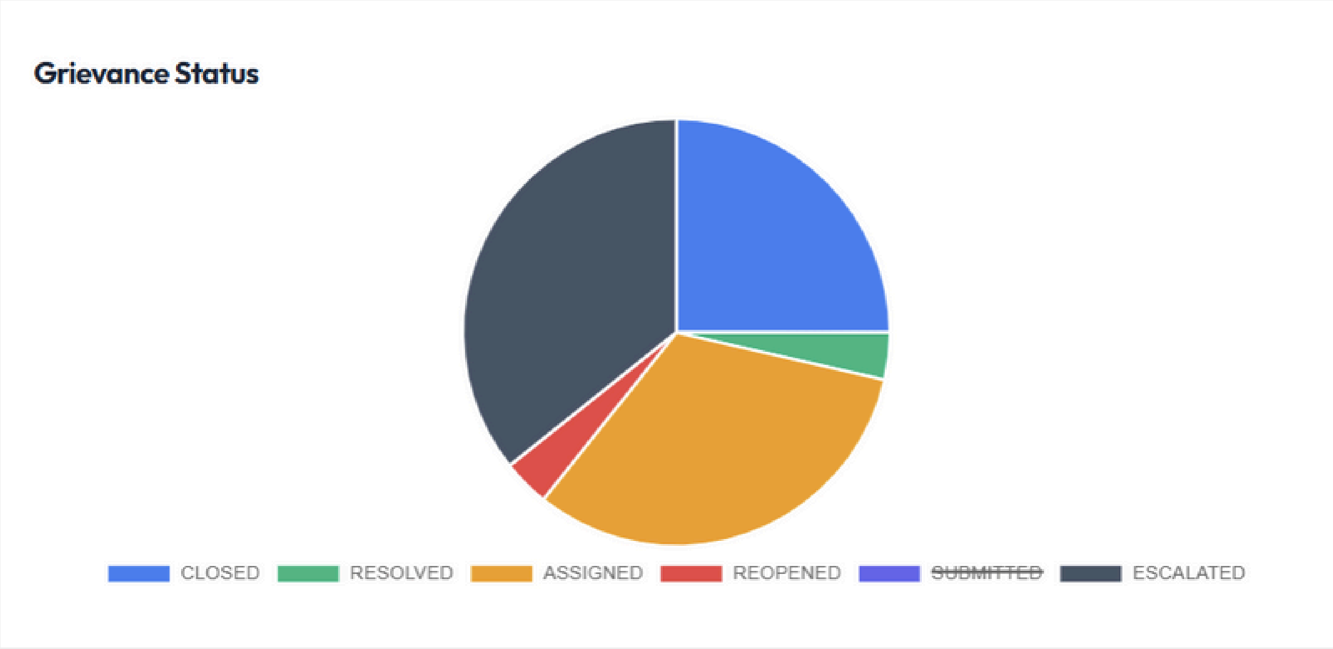
RESOLVED 8

GRIEVANCE RESOLUTION RATE

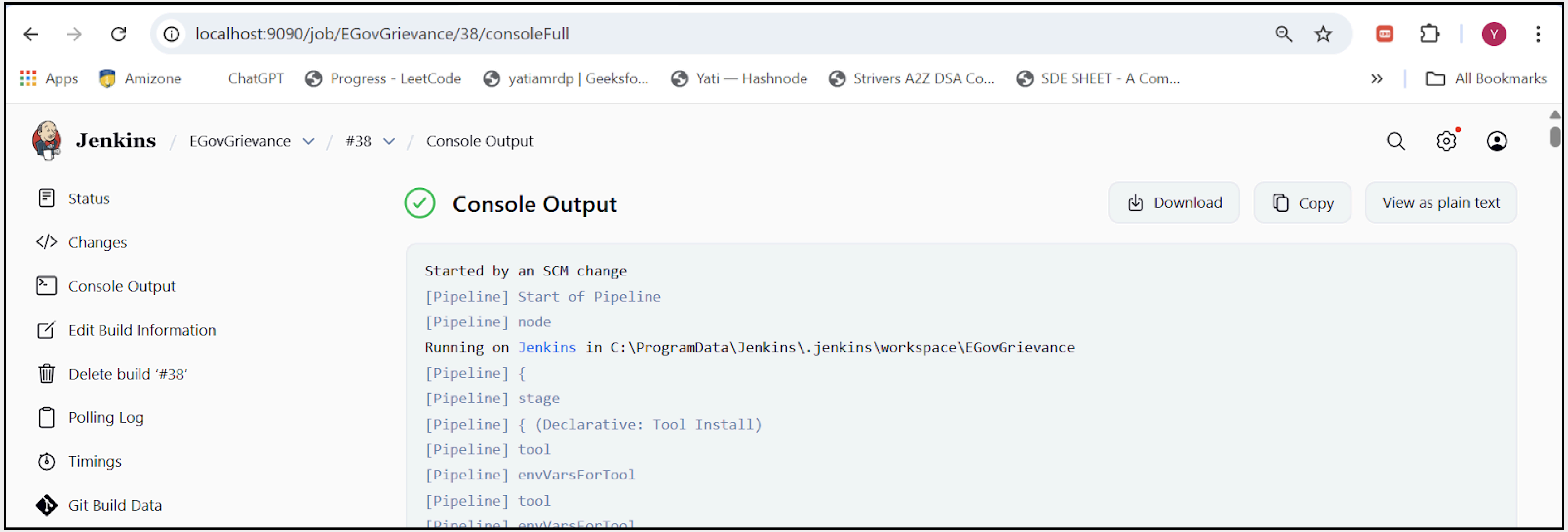
27%

Resolved vs Total



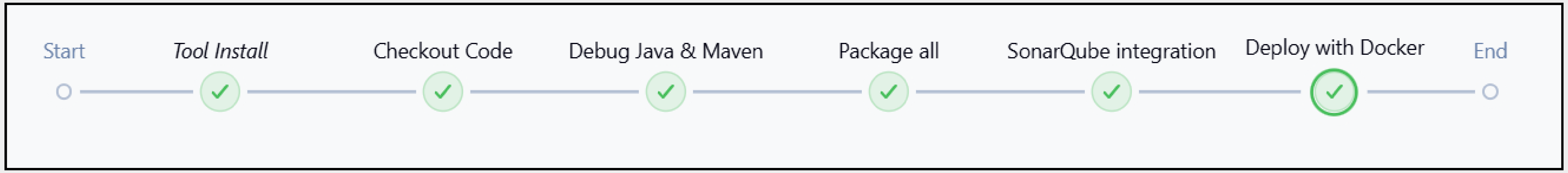


Jenkins (using SCM , Manual Build)

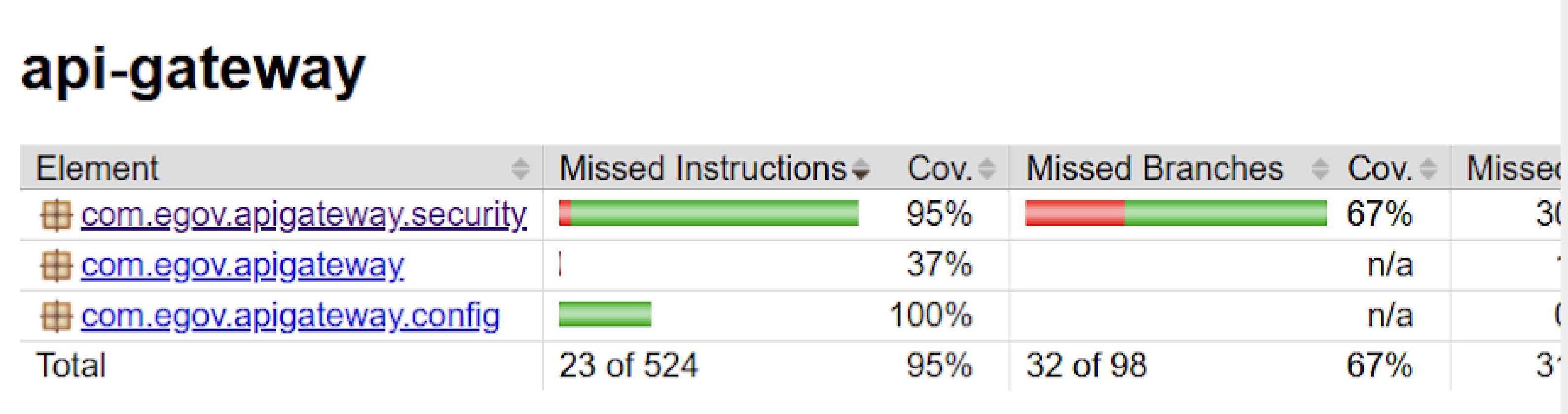


Console Output

```
Started by an SCM change
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in C:\ProgramData\Jenkins\.jenkins\workspace\EGovGrievance
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Declarative: Tool Install)
[Pipeline] tool
[Pipeline] envVarsForTool
[Pipeline] tool
[Pipeline] envVarsForTool
```

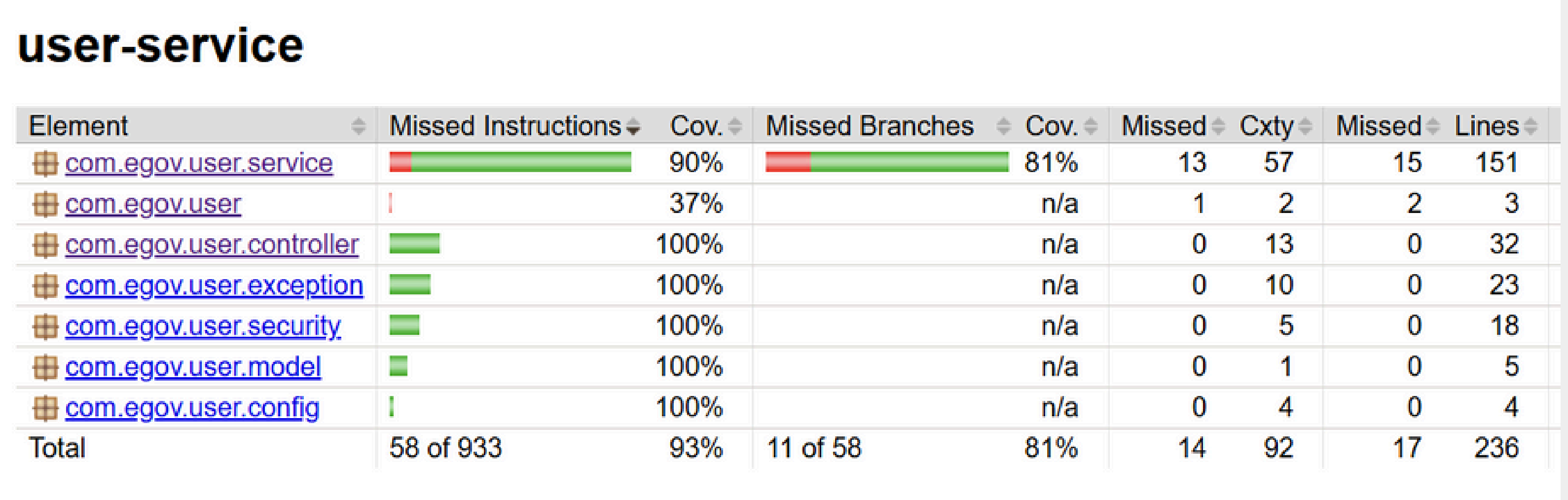


JacocoReports



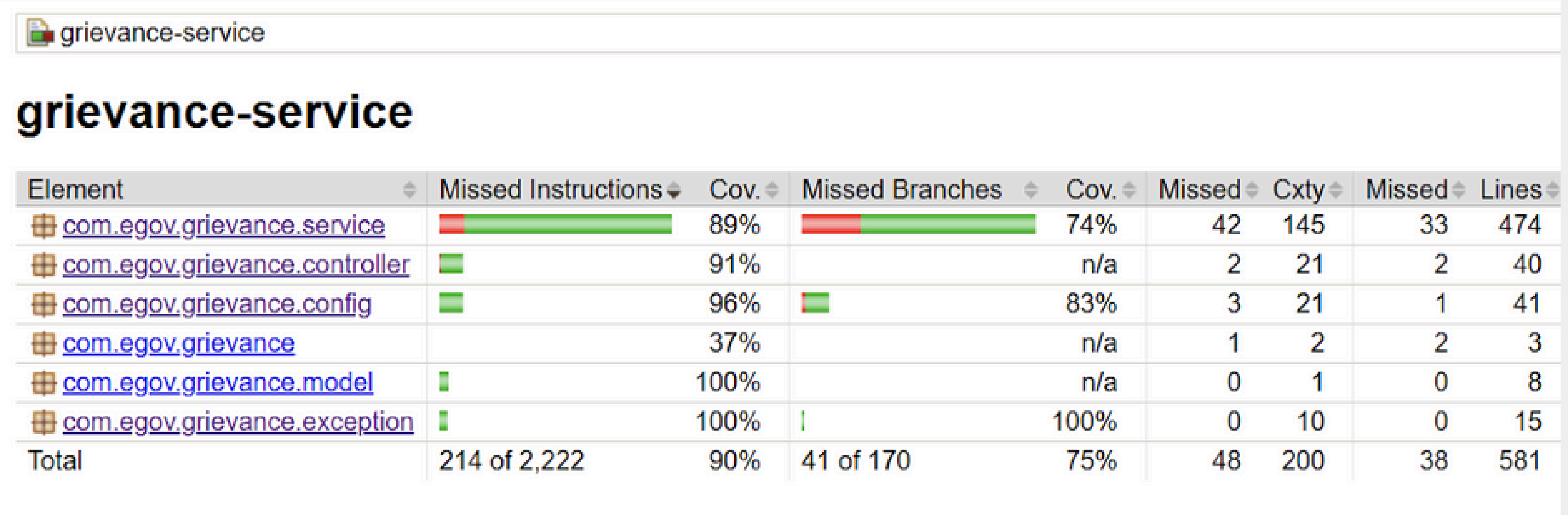
api-gateway

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed
com.egov.apigateway.security		95%		67%	30
com.egov.apigateway		37%		n/a	1
com.egov.apigateway.config		100%		n/a	0
Total	23 of 524	95%	32 of 98	67%	31



user-service

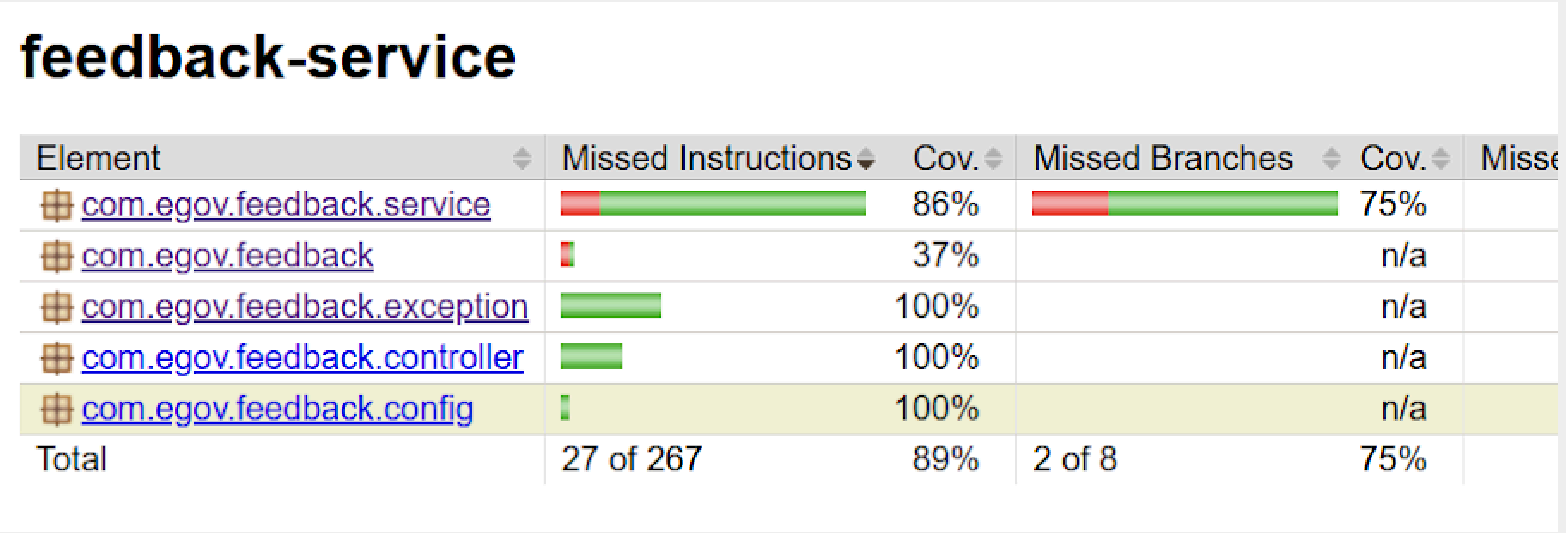
Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty	Missed	Lines
com.egov.user.service		90%		81%	13	57	15	151
com.egov.user		37%		n/a	1	2	2	3
com.egov.user.controller		100%		n/a	0	13	0	32
com.egov.user.exception		100%		n/a	0	10	0	23
com.egov.user.security		100%		n/a	0	5	0	18
com.egov.user.model		100%		n/a	0	1	0	5
com.egov.user.config		100%		n/a	0	4	0	4
Total	58 of 933	93%	11 of 58	81%	14	92	17	236



grievance-service

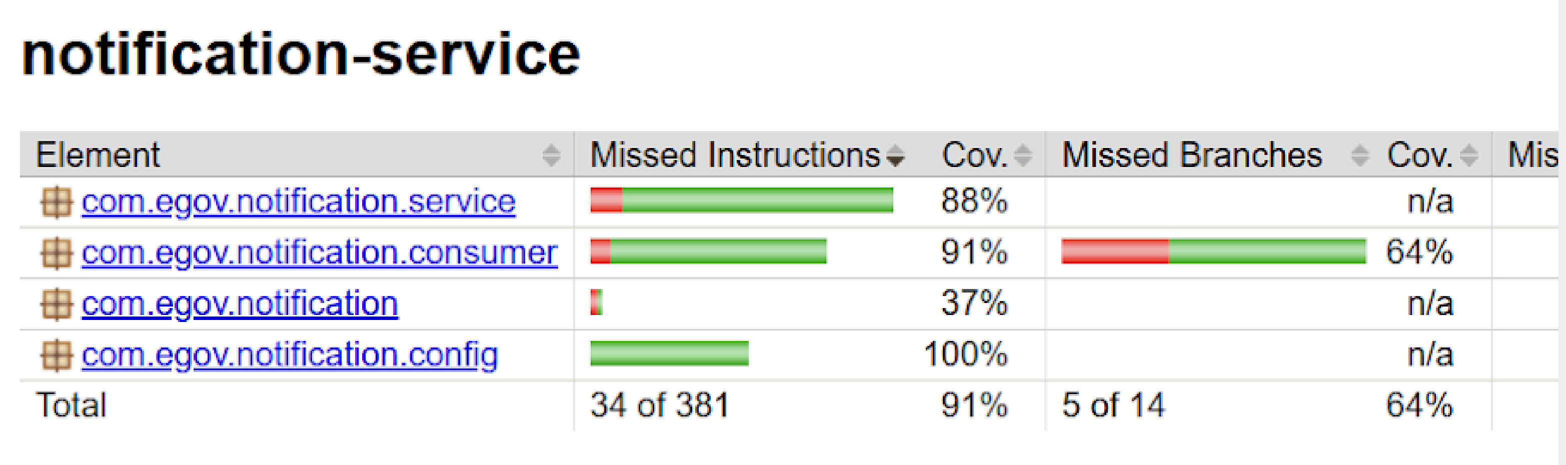
grievance-service

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty	Missed	Lines
com.egov.grievance.service		89%		74%	42	145	33	474
com.egov.grievance.controller		91%		n/a	2	21	2	40
com.egov.grievance.config		96%		83%	3	21	1	41
com.egov.grievance		37%		n/a	1	2	2	3
com.egov.grievance.model		100%		n/a	0	1	0	8
com.egov.grievance.exception		100%		100%	0	10	0	15
Total	214 of 2,222	90%	41 of 170	75%	48	200	38	581



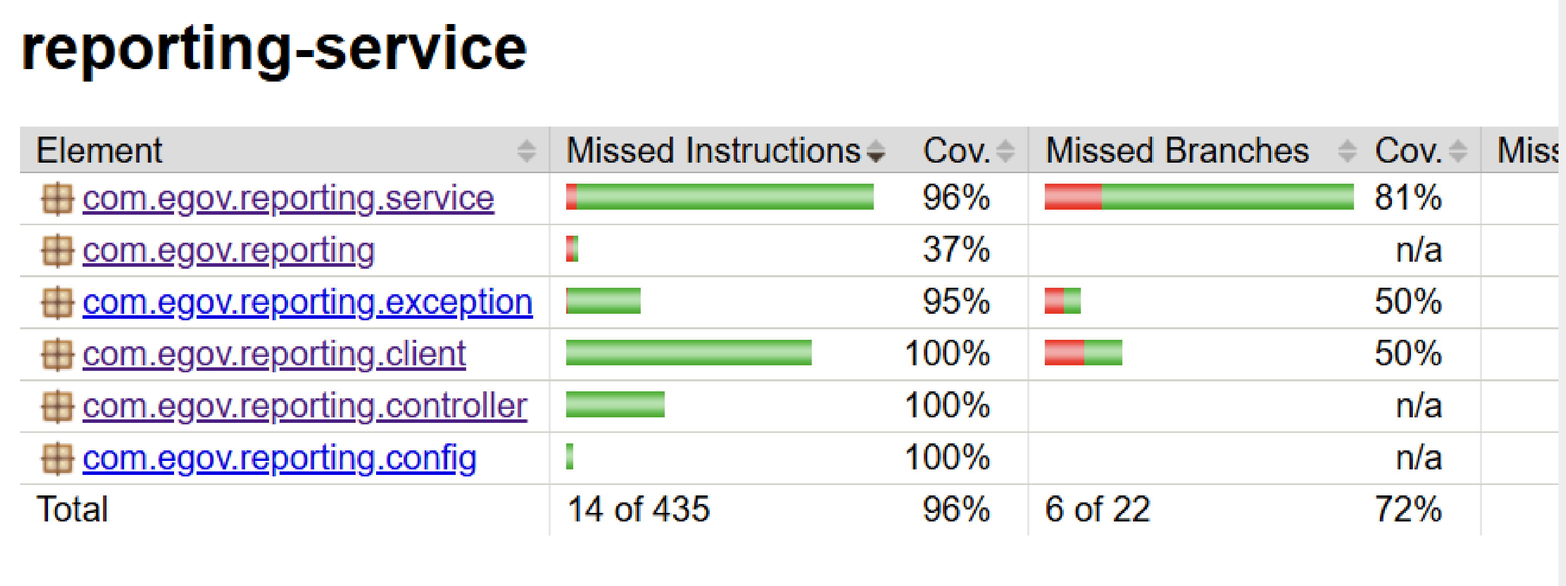
feedback-service

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Misse
com.egov.feedback.service		86%		75%	
com.egov.feedback		37%		n/a	
com.egov.feedback.exception		100%		n/a	
com.egov.feedback.controller		100%		n/a	
com.egov.feedback.config		100%		n/a	
Total	27 of 267	89%	2 of 8	75%	



notification-service

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Mis
com.egov.notification.service		88%		n/a	
com.egov.notification.consumer		91%		64%	
com.egov.notification		37%		n/a	
com.egov.notification.config		100%		n/a	
Total	34 of 381	91%	5 of 14	64%	



reporting-service

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Miss
com.egov.reporting.service		96%		81%	
com.egov.reporting		37%		n/a	
com.egov.reporting.exception		95%		50%	
com.egov.reporting.client		100%		50%	
com.egov.reporting.controller		100%		n/a	
com.egov.reporting.config		100%		n/a	
Total	14 of 435	96%	6 of 22	72%	

Conclusion

What the System Achieves:

- ✓ Centralized grievance tracking
- ✓ Clear ownership and accountability
- ✓ SLA-based escalation without schedulers
- ✓ Full audit trail
- ✓ Citizen transparency
- ✓ Scalable and maintainable architecture

Thank you!