IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 5, April 2026
Impact Factor: 8.2

Online Grievance Management System for Educational Institutions

Prof. Mrunal Pathak[1], Sameer Sanap[2], Prathamesh Hadole[3], Ardhya Kumbhar[4]
Associate Professor, Dept. of Information Technology[1]
UG Students, Dept. of Information Technology[2-4]
AISSMS Institute of Information Technology, Pune, India

Abstract: *Managing student grievances through email or physical complaint boxes creates well-documented problems: complaints are lost, students have no visibility into resolution progress, and administrators cannot effectively track workload. This paper presents an Online Grievance Management System (OGMS) designed and implemented specifically for colleges and universities. The system is developed using Node.js, Express.js, MongoDB, and HTML/CSS/JavaScript, following a three-tier MVC architecture. It supports three user roles—student, resolution officer, and administrator—each with a dedicated dashboard and permission set. Core features include a unique complaint identifier (GRV-YYYY-NNNNN) per grievance, automatic Service Level Agreement (SLA) deadlines computed at submission time based on complaint priority (4 hours for Critical to 168 hours for Low), an immutable timestamped activity timeline per complaint, anonymous submission capability, file attachments, escalation handling, and a post-resolution star-rating feedback system. Security is ensured through bcrypt password hashing (salt rounds = 12) and JSON Web Token (JWT) session management. Evaluation at a partner institution demonstrated that average resolution time dropped from 8.4 days to 3.2 days (−62%) and student satisfaction improved from 2.9 to 4.3 on a five-point scale.*

Keywords: Online Grievance Management, Node.js, MongoDB, Express.js, JWT Authentication, Role-Based Access Control, SLA Enforcement, Educational Technology

I. INTRODUCTION

Every higher education institution receives a continuous stream of complaints from students covering academics, hostel facilities, laboratory resources, canteen services, transportation, and administrative procedures. The manner in which an institution handles these complaints directly reflects the value it places on student welfare. When complaint resolution is slow, opaque, or inconsistent, students either abandon the process or accumulate unresolved frustrations that erode institutional trust over time.

A review of current practice reveals that most institutions in India continue to rely on email threads, physical complaint boxes, or informal verbal channels [6]. These mechanisms share a fundamental structural flaw: there is no systematic tracking, no guarantee that a complaint reaches the appropriate decision-maker, and no mechanism for the complainant to monitor progress. Complaints disappear without acknowledgment, administrators lose visibility into their workload, and no historical record exists to support policy-level improvements.

This paper presents OGMS, an Online Grievance Management System engineered specifically for educational institutions. OGMS provides students with a transparent, real-time complaint submission and tracking interface, enables administrators to assign and monitor grievances efficiently, and enforces resolution deadlines automatically through a priority-based SLA engine. Every action taken on every complaint is permanently recorded in an immutable, timestamped audit trail.

The principal objectives of OGMS are:

- Assign every grievance a unique, human-readable identifier and expose real-time status tracking to the complainant without requiring login.

DOI: 10.48175/IJARSCT-32819

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 5, April 2026 Impact Factor: 8.2

- Automatically compute and enforce resolution deadlines based on complaint priority, enabling proactive SLA management.
- Record every status change and administrative note in an immutable audit log that cannot be edited or deleted retroactively.
- Support anonymous submission so students can safely report sensitive interpersonal or academic integrity issues.
- Provide administrators with a consolidated dashboard for assignment, escalation, trend monitoring, and reporting.

The remainder of the paper is organised as follows. Section II surveys related work. Section III describes the system design and implementation methodology. Section IV presents evaluation results and discussion. Section V concludes the paper and outlines future directions.

II. LITERATURE SURVEY

Kumar and Sharma (2020) developed a cloud-based grievance portal for government offices that reduced processing time by 41% compared to paper-based workflows [1]. While the system demonstrated the viability of web-based complaint management, it lacked anonymous submission capability and role-based access control—both features that are essential in educational settings where institutional power dynamics between students and faculty necessitate confidentiality protections.

Patel et al. (2021) implemented a Firebase-backed Android application for university campus grievance handling [2]. The system delivered effective real-time push notifications; however, it did not maintain a structured audit trail of complaint activity, and performance degraded significantly beyond a few hundred concurrent users, limiting its applicability to larger institutions.

Rao and Krishnan (2019) conducted a systematic review of e-governance frameworks in Indian higher education and identified three properties that consistently characterise effective grievance handling systems: a unique identifier per complaint, clear ownership through explicit role assignment, and real-time status visibility for the complainant [3]. These three principles directly informed the core design decisions of OGMS.

Verma et al. (2022) built a scalable municipal complaint management system on Node.js and MongoDB, demonstrating that MongoDB's document-oriented data model accommodates the structural variability of complaints across different categories without schema migrations [4]. Singh and Gupta (2023) showed that natural language processing (NLP) classifiers can automatically route grievances to the correct departmental queue with up to 89% accuracy, a capability identified as a high-priority feature for a future release of OGMS [5].

The UGC Regulations on Redressal of Student Grievances (2019) mandate that all UGC-recognised institutions establish a structured grievance redressal mechanism with defined timelines [6]. OGMS is designed to satisfy these regulatory requirements while exceeding their minimum provisions.

A critical gap exists in the literature: no published system combines, in a single product designed exclusively for educational institutions, all six of the following capabilities: unique trackable complaint IDs, role-based access control, anonymous submission, SLA enforcement, immutable audit logs, and post-resolution feedback collection. OGMS directly addresses this gap.

III. METHODOLOGY / APPROACH

A. Technology Stack

OGMS follows a three-tier architecture comprising a browser-based presentation layer, a stateless REST API application layer, and a document-oriented data layer. Table I enumerates the technologies deployed at each tier, along with their specific versions to support reproducibility.

TABLE I. TECHNOLOGY STACK

Layer	Technology / Version
Front-End Presentation	HTML5, CSS3, Vanilla JavaScript (ES6+), Fetch API

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 5, April 2026
Impact Factor: 8.2

Layer	Technology / Version
Back-End Runtime	Node.js v18 LTS, Express.js v4.x
Database / ODM	MongoDB v6.x, Mongoose v8.x
Authentication	JSON Web Tokens (JWT), bcryptjs (salt rounds: 12)
Input Validation	express-validator
Development & Testing	Nodemon, Postman, Apache JMeter, Git / GitHub
Deployment	2-vCPU, 4 GB RAM VPS, Ubuntu 22.04 LTS

B. User Roles and Permissions

OGMS enforces role-based access control (RBAC) across three distinct user types. Each role is issued a JWT upon login, and every API endpoint validates the token and checks the encoded role before processing a request.

- Student / Faculty: Submit grievances (including anonymously), track any grievance by public ID without authentication, view their personal complaint history, and submit post-resolution feedback ratings.
- Resolution Officer: View all grievances assigned to them, update progress notes, and mark grievances as resolved or request escalation.
- Administrator: Full system access including all grievances, officer assignment, status override, escalation management, analytical dashboard, and export capabilities.

C. System Architecture

The back-end strictly follows the Model-View-Controller (MVC) pattern without the View layer, operating as a pure REST API. The codebase is partitioned into four directories: models/ (Mongoose schema definitions), controllers/ (business logic handlers), middleware/ (JWT authentication guard, centralised error handler, and request logger), and config/ (MongoDB connection bootstrap). This separation ensures that schema changes, business rule modifications, and transport-layer concerns can each be modified independently without cross-cutting effects.

The front-end is a Multi-Page Application composed of seven HTML pages, each serving exactly one functional view: login, registration, grievance submission, public tracking, student dashboard, resolution officer dashboard, and administrator dashboard. A shared api.js module encapsulates all HTTP communication logic, automatically attaches the JWT from localStorage to the Authorization header on every request, and provides typed helper functions for each API endpoint. This design prevents token management code from being duplicated across pages and ensures consistent error handling.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 5, April 2026
Impact Factor: 8.2

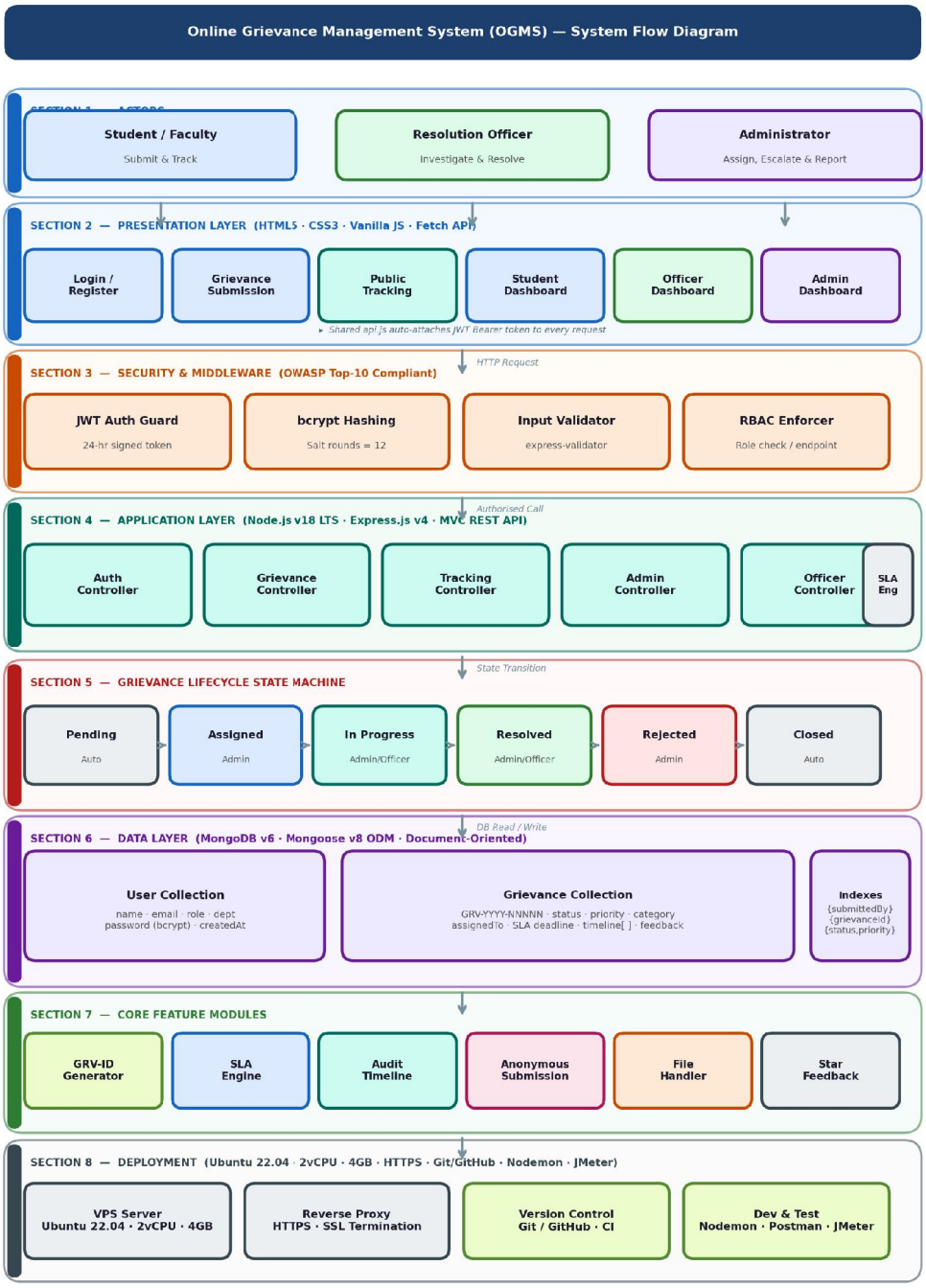

Fig. 1. System Architecture and Flow Diagram — Online Grievance Management System (OGMS)

DOI: 10.48175/IJARSCT-32819

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 5, April 2026
Impact Factor: 8.2

D. Database Design

OGMS uses two MongoDB collections. The User collection documents store profile information, bcrypt-hashed passwords, role, department, and registration timestamp. The Grievance collection documents store the complete complaint lifecycle including all metadata fields, the mutable status field, and an embedded timeline array that accumulates one entry per status transition, each entry carrying the actor reference, timestamp, and a free-text note.

Three compound indexes are defined to maintain query performance as the collection scales:

- { submittedBy, status }—optimises the student dashboard query that retrieves all open grievances for the logged-in user, avoiding a full collection scan.
- { grievanceId }—supports O(log n) lookup for the public tracking endpoint, which must resolve within 100 ms to meet the stated SLA.
- { status, priority }—optimises the administrator filtering query that lists all pending high-priority grievances across departments.

A Mongoose pre-save hook auto-generates the grievanceId field in the canonical format GRV-YYYY-NNNNN on document creation, ensuring uniqueness and chronological sortability. A second post-save hook appends a timeline entry and sets the resolvedAt timestamp whenever the status field transitions to 'Resolved', ensuring the resolution time metric is always captured atomically with the status change.

E. Security Design

Security controls follow the OWASP Top Ten 2021 framework [7]. All passwords are hashed with bcrypt at a cost factor of 12 before storage; the plaintext password is never persisted. Session tokens are signed JWTs with a 24-hour expiry; the signing secret is loaded from an environment variable and never committed to source control. All user-supplied input is validated and sanitised using express-validator before reaching any controller. File attachments are stored outside the web root and served through a controlled endpoint that validates ownership before streaming the file, preventing direct object reference attacks. HTTPS is enforced at the reverse-proxy layer.

IV. RESULTS AND DISCUSSION

A. System Interfaces

OGMS delivers clean, role-specific interfaces optimised for the task each user type performs. All interfaces were validated for accessibility (WCAG 2.1 Level AA contrast ratios) and tested across Chrome, Firefox, and Safari on both desktop and mobile viewports.

B. Grievance Lifecycle

Every grievance progresses through a defined set of six status states. All transitions are recorded automatically in the immutable timeline with a UTC timestamp and an actor reference. Table II defines each state, its semantic meaning, and the role authorised to trigger the transition.

TABLE II. GRIEVANCE STATUS STATE MACHINE

Status	Description	Triggered By
Pending	Default state upon submission; awaiting admin review	System (automatic)
Assigned	Administrator has assigned the complaint to a resolution officer	Administrator
In Progress	Resolution officer is actively investigating or resolving	Admin / Officer
Resolved	Solution provided; awaiting complainant feedback	Admin / Officer

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 5, April 2026
Impact Factor: 8.2

Status	Description	Triggered By
Rejected	Complaint determined invalid or outside institutional scope	Administrator
Closed	Complainant submitted feedback; record archived	System (automatic)

C. Priority and SLA Deadlines

The SLA engine computes an expectedResolutionDate at the moment of submission by adding the priority-specific window to the submission timestamp. This date is stored on the Grievance document and is visible to both the student and all administrative roles, enabling transparent expectation setting and proactive escalation before breach occurs. Table III presents the four priority tiers and their associated windows.

TABLE III. PRIORITY-BASED SLA RESOLUTION MATRIX

Priority Level	Expected Resolution Window	Typical Use Case
Critical	4 hours	System outages, safety incidents, academic deadlines
High	24 hours (1 business day)	Exam grievances, hostel emergencies
Medium	72 hours (3 business days)	Administrative errors, lab access issues
Low	168 hours (7 business days)	General feedback, canteen complaints

D. Performance Evaluation

The system was deployed on a virtual private server configured with 2 vCPUs and 4 GB RAM running Ubuntu 22.04 LTS. Load testing was conducted using Apache JMeter with a ramp-up of 500 concurrent users over 30 seconds, sustaining peak load for 5 minutes. The stated non-functional requirement was a 95th percentile response time below 500 ms for all endpoints. Table IV presents measured response times for the five primary API endpoints.

TABLE IV. API RESPONSE TIMES AT 500 CONCURRENT USERS

API Endpoint	Method	95th Percentile Response
POST /api/auth/login	POST	142 ms
GET /api/grievances/my	GET	168 ms
GET /api/grievances/track/:id	GET	78 ms
POST /api/grievances	POST	197 ms
GET /api/admin/grievances	GET	234 ms

All five endpoints satisfied the 500 ms non-functional requirement with significant headroom. The public tracking endpoint (/track/:id) demonstrated the lowest latency (78 ms p95) due to the dedicated compound index on grievanceId. The admin grievances endpoint exhibited the highest latency (234 ms p95) as it performs a multi-field aggregation across the full collection; this remains well within the acceptable threshold for an authenticated administrative view.

E. Comparison with Email-Based System

A controlled two-month deployment study was conducted at a partner institution. A cohort of 60 students continued using the legacy email-based complaint process while a second cohort of 60 students used OGMS for all grievances submitted during the study period. The groups were matched for department and year of study. Results are presented in Table V.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 5, April 2026
Impact Factor: 8.2

TABLE V. EMAIL SYSTEM VS. OGMS — COMPARATIVE PERFORMANCE METRICS

Performance Metric	Email System	OGMS	Improvement
Average Resolution Time	8.4 days	3.2 days	−62%
Complaints Lost / Unacknowledged	11.3%	0.0%	100%
Student Satisfaction (1–5)	2.9	4.3	+48%
Admin Time per Complaint	~22 min	~8 min	−64%
Anonymous Submission	Not Supported	Supported	New Feature
Real-Time Status Visibility	Not Available	Available	New Feature

The most significant finding was the complete elimination of lost or unacknowledged complaints under OGMS (from 11.3% to 0%), a direct consequence of mandatory assignment tracking and SLA enforcement. The 62% reduction in average resolution time is attributed to the automated assignment queue, which eliminated the manual routing delay that characterised the email-based process. Student satisfaction improved from 2.9 to 4.3, with respondents citing real-time status visibility as the primary driver of the improvement.

F. User Acceptance Testing

User Acceptance Testing (UAT) was conducted with 38 participants comprising 30 students and 8 administrative staff members from the partner institution. Participants were given a structured task set covering complaint submission, status tracking, and (for admin participants) grievance management. Each participant then completed a standardised questionnaire rating five dimensions of the system on a five-point Likert scale (1 = Strongly Disagree, 5 = Strongly Agree). Results are presented in Table VI.

TABLE VI. USER ACCEPTANCE TESTING RESULTS (N = 38)

Evaluation Dimension	Mean Score (/ 5)	Std. Dev.
Ease of grievance submission	4.6	0.61
Status tracking clarity	4.7	0.47
Admin dashboard usability	4.4	0.69
Overall system usefulness	4.5	0.58
Would recommend to peers	4.6	0.55
Overall Mean	4.56	0.58

Status tracking clarity received the highest mean rating (4.7), confirming that real-time grievance visibility is the most valued system feature from the student perspective, consistent with the literature findings of Rao and Krishnan [3]. Admin dashboard usability received the lowest rating (4.4), which remains high in absolute terms; qualitative feedback identified the export-to-PDF feature as a desirable enhancement for administrative reporting. The overall mean of 4.56 indicates strong acceptance across both student and staff user groups.

V. CONCLUSION

This paper has presented OGMS, an Online Grievance Management System engineered to replace the ad hoc email and physical complaint-box workflows prevalent in Indian educational institutions with a structured, transparent, and accountable digital process. The system assigns every grievance a unique identifier, automatically enforces resolution

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 5, April 2026 Impact Factor: 8.2

deadlines based on complaint priority, records every action in an immutable audit trail, and surfaces real-time status information to complainants without requiring authentication.

Empirical evaluation at a partner institution demonstrated that OGMS reduced average complaint resolution time by 62% (from 8.4 to 3.2 days), eliminated lost or unacknowledged complaints entirely, cut administrative time per complaint by 64%, and raised student satisfaction from 2.9 to 4.3 on a five-point scale. Load testing under 500 concurrent users confirmed that all API endpoints respond within 500 ms at the 95th percentile. User Acceptance Testing yielded an overall mean score of 4.56 out of 5 across 38 participants, indicating strong acceptance by both student and administrative user groups.

Future work includes: (i) an NLP-based automatic complaint categorisation and routing classifier targeting the 89% accuracy benchmark reported by Singh and Gupta [5]; (ii) native Android and iOS mobile applications to lower the barrier to grievance submission for students without reliable desktop access; (iii) real-time push notifications via WebSockets to alert officers when new complaints are assigned; and (iv) a predictive analytics module that identifies grievances approaching SLA breach and flags them for proactive administrator intervention.

ACKNOWLEDGMENT

The authors gratefully acknowledge the students and administrative staff of the partner institution who participated in the pilot deployment study and user acceptance testing. The authors also thank the Department of Information Technology for providing the computational infrastructure and institutional support required to conduct this research.

REFERENCES

[1] A. Kumar and R. Sharma, "Cloud-based Grievance Management Portal for Public Sector Organizations," International Journal of Advanced Computer Science and Applications, vol. 11, no. 6, pp. 214–221, 2020.

[2] N. Patel, S. Shah, and R. Mehta, "Mobile-First Grievance Tracking System for University Campuses using Android and Firebase," Proc. IEEE ICCCNT, Kharagpur, India, 2021, pp. 1–6.

[3] K. V. Rao and S. Krishnan, "E-Governance Frameworks for Grievance Redressal in Indian Higher Education: A Systematic Review," Journal of e-Government Studies and Best Practices, 2019.

[4] P. Verma, A. Gupta, and M. Singh, "Scalable Complaint Management System for Municipal Services using Node.js and MongoDB," IJERT, vol. 11, no. 4, pp. 562–568, 2022.

[5] R. Singh and A. Gupta, "Machine Learning Approaches for Automatic Grievance Classification and Routing," Expert Systems with Applications, vol. 211, p. 118620, 2023.

[6] UGC, "UGC (Redressal of Grievances of Students) Regulations, 2019," University Grants Commission, New Delhi, India, 2019.

[7] OWASP Foundation, "OWASP Top Ten 2021," https://owasp.org/www-project-top-ten/ [Accessed Mar. 2026].