Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 10 , October - 2026

SmartMandya: An Improved Digital Gram Panchayat Platform for Citizen Services, Application Tracking, and Transparent Rural Governance

[Manjunath M.k][1], [Darshan Gowda S][1], [Roopini K.S][1] , [Ramya C.N][1], [Guide / Co-author Ms.Chandana.S][1]
[1][Department of Computer Science and Engineering], [*G Madegowda Institute of Technology*], [*Bharathinagar,*

Abstract*— Gram Panchayats are an important point of contact between rural residents and local public services. However, citizens may face difficulties in finding service information, submitting requests, checking application progress, and reaching the appropriate office. This paper analyses SmartMandya, a web-based Gram Panchayat project, using its supplied source code and a review of four related IEEE papers on AI-enabled e-governance, blockchain-supported financial transparency, digital public administration, and sentiment analysis. The inspected implementation contains multiple information pages, citizen registration and login interfaces, service-application workflows, application-status management, an administrative interface, and a contact form supported by a Node.js/Express backend and JSON-file storage. The analysis also identifies limitations that should be addressed before the platform is treated as a secure production system, including client-side administrator credentials, plain-text password storage in the inspected model, file-based persistence, and incomplete integration of some dashboard functions. The paper proposes a phased improvement plan covering secure authentication, role-based authorization, database-backed records, Kannada-English access, accessible mobile-first design, grievance tracking, auditable service records, and optional future AI-assisted feedback analysis. Blockchain and AI are presented as future extensions rather than existing features. No performance or user-study results are claimed because a controlled evaluation dataset was not supplied.

***Keywords*—** *Gram Panchayat; e-governance; rural digital services; citizen applications; SmartMandya; transparency; web application; security*

I. INTRODUCTION

Digital public-service portals can reduce the need for repeated office visits by making service information and request status available online. At the Gram Panchayat level, a useful portal should be easy to understand, work on common mobile devices, support residents with different levels of digital literacy, and provide reliable records for staff. The objective of the SmartMandya project is to provide a web-based access point for Gram Panchayat information and selected citizen-facing workflows.

The supplied project source contains public information pages for the Panchayat, services, initiatives, and contact details, alongside citizen login/registration, service applications, administrative application review, and a dashboard-oriented interface. The implementation uses HTML, CSS, Bootstrap, JavaScript, and a Node.js/Express server. The present study distinguishes features visible in the supplied code from recommendations that have not yet been implemented or tested.

The contribution of this paper is threefold: (1) document the current project structure and existing workflows; (2) relate the project to research on digital governance, blockchain-supported transparency, and citizen feedback analysis; and (3) define a practical improvement roadmap, including security controls and an evaluation plan suitable for a student project.

II. RELATED WORK

A. AI and Blockchain in E-Governance

Ajay et al. discuss the potential of combining artificial intelligence and blockchain in smart-city e-governance. Their paper presents AI as a way to analyse information and support decisions, while blockchain can support integrity and confidence in digital transactions [1]. For SmartMandya, these ideas suggest future possibilities such as analysing recurring service complaints and maintaining tamper-evident records of important administrative events. They do not imply that AI or blockchain is already present in the current code.

B. Blockchain for Rural Public-Fund Transparency

Mainali et al. propose a blockchain-based e-governance approach for transparent allocation and utilization of local-government funds in rural regions [2]. The work is directly relevant to Panchayat governance because public works and funding records need traceability. A practical first step for SmartMandya would be to publish verified project information, approved amounts, work status, and dated updates with an audit trail. A permissioned blockchain could be evaluated later if multiple authorized organizations need shared, tamper-evident records; it should not be adopted without a clear operational need.

C. Sentiment Analysis for E-Government Feedback

Nagesh et al. examine a dual approach to e-government feedback analysis using text and facial-expression recognition [3]. The study highlights the potential of structured feedback analysis, while also raising privacy and consent considerations around facial data. For a small Gram Panchayat project, text-based analysis of voluntarily submitted complaints is a lower-risk initial option. Facial recognition is not proposed as a necessary feature because it would introduce additional privacy, consent, fairness, and data-protection burdens.

D. Digital Public Governance and Inclusion

Brechko et al. discuss digital public governance in the context of digitalization and demographic challenges, including population ageing, migration, regional differences, service continuity, and digital literacy [4]. These themes support a design that is simple, mobile-friendly, accessible, and usable by people who may need assistance. The Government of

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 10 , October - 2026

India's eGramSwaraj description also identifies Panchayat profile, planning, progress reporting, accounting, asset directory, and user management as important digital-governance modules [5]. SmartMandya can use these systems as a reference point without claiming official integration.

III. ANALYSIS OF THE EXISTING SMARTMANDYA PROJECT

A. Technology and Structure

Fig. 3. SmartMandya Functional Modules

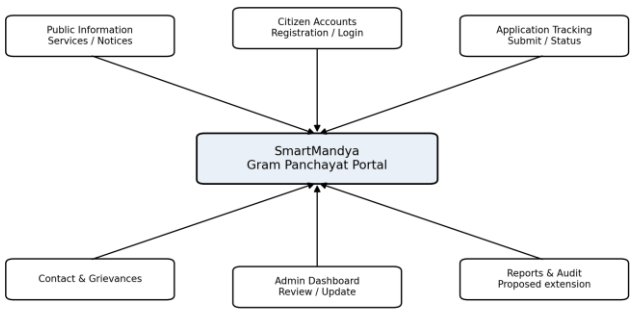


Fig. 1. Functional modules in the current/proposed SmartMandya portal.

The supplied archive includes HTML pages, CSS, Bootstrap assets, client-side JavaScript, and a Node.js/Express server. The server exposes API routes for citizen registration, login, service application submission, a citizen's application list, administrative application listing and status updates, and contact-form submissions. Application and citizen records are stored in JSON files under the project data directory. The public deployment was reachable during review, but the fetched page content exposed only a title/short landing response; therefore, the live deployment was not treated as proof that every local-code feature works in production.

B. Existing Features Found in the Supplied Code

- Public information pages: home, About Us, Services, Initiatives, and Contact.
- Service catalogue covering certificates, property tax, water and sanitation, roads, agricultural support, social welfare, education and health, public distribution, and complaint registration.
- Citizen registration and login API endpoints, with a workflow to submit a service application.
- Application status fields supporting pending, approved, and rejected states, plus administrative remarks and timestamps.
- Administrative application listing with totals for pending, approved, and rejected applications.
- Contact form endpoint that records a citizen's name, phone, optional email/address, subject, message, urgency flag, and submission timestamp.
- Dashboard and user/yojana-management pages are present, but some of their functions use local PHP URLs or separate interface logic and require end-to-end integration testing.

C. Security and Reliability Findings

The source-code inspection found several issues that should be resolved before real citizen information is stored. The administrator login page contains a fixed username and password in browser-side JavaScript, which can be discovered by anyone who inspects the page. The application model stores citizen passwords as plain text and compares them directly during login. The server enables broad CORS access, and the inspected administrative API endpoints do not show a server-side authentication/authorization check. JSON files are used as the primary store, which can become unreliable with concurrent writes, backups, multiple server instances, or platform restarts. These are code-review findings, not claims that an external penetration test has been performed.

The project also contains references to local PHP endpoints for some yojana-management actions. Those references may not work in the deployed environment unless the corresponding PHP service is configured. The live site's limited visible response during this review means that deployment behaviour should be verified separately from local source-code behaviour.

IV. PROPOSED IMPROVEMENTS

A. Security and Data Management

- Replace browser-side administrator credentials with server-side authentication and remove all hard-coded secrets.
- Hash passwords using a well-established password-hashing algorithm such as Argon2id or bcrypt; never store raw passwords.
- Introduce session-based authentication or securely configured tokens, with role-based access for citizens, staff, and administrators.
- Protect administrative APIs with authorization checks on every request, validate inputs on the server, add rate limiting, and configure CORS only for trusted origins.
- Move citizen and application records from JSON files to a relational database such as PostgreSQL or MySQL, with backups, migrations, and appropriate access controls.
- Use HTTPS in deployment, keep secrets in environment variables, maintain audit logs, and define data retention and deletion practices.

B. Citizen Experience and Inclusion

- Provide a Kannada-first interface with an English option, using clear labels and short instructions.
- Improve mobile responsiveness, keyboard navigation, contrast, form error messages, and screen-reader labels.
- Offer a simple application reference number and a status page showing submission date, current status, remarks, and next steps.
- Create a grievance workflow with category, optional attachment, priority, assigned officer, response deadline, and resolution history.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 10 , October - 2026

- Make service eligibility, required documents, office hours, contact information, and official links easy to find.
- Design low-bandwidth pages and allow assisted/offline submission by authorized staff where practical, with clear consent and recordkeeping.

C. Transparency and Governance

- Add a public notice board for Gram Sabha meetings, deadlines, approved initiatives, and service announcements.
- Publish verified public-works information such as work title, ward, approved amount, planned dates, current status, and dated progress updates.
- Maintain an append-only administrative audit trail for changes to application status and published public records.
- Provide basic dashboard summaries for submitted applications, pending cases, average resolution time, and service-category counts, while avoiding exposure of personal data.
- Consider a permissioned blockchain only if a defined multi-party audit requirement cannot be met adequately with signed database logs and conventional controls.

D. Optional AI-Assisted Feedback Analysis

A later version could classify voluntary text feedback into broad categories such as water supply, roads, certificates, sanitation, or welfare services, and estimate sentiment to help staff identify recurring concerns. The system should display AI output as decision support rather than as a final decision, allow staff correction, retain a route for human review, and document model limitations. Personal identifiers should be removed where possible. Facial-expression analysis is intentionally excluded from the initial proposal because it is not required to deliver the core service workflow and would increase privacy risks.

V. PROPOSED SYSTEM ARCHITECTURE AND WORKFLOW

The improved system can be organized into five layers: (1) a responsive citizen interface; (2) an authenticated application/API layer; (3) business services for registration, applications, grievances, notices, and reporting; (4) a relational data store with backups and audit records; and (5) a monitoring and reporting layer for authorized staff. External government services should be linked or integrated only where an official API and authorization are available.

Proposed citizen workflow:

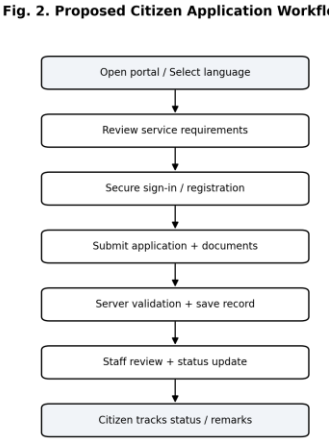


Fig. 2. Proposed citizen application workflow.

1) Citizen opens the portal and chooses Kannada or English.
2) Citizen reviews service requirements and registers or signs in securely.
3) Citizen submits a service request and receives a unique application reference.
4) The server validates the request, stores it, and records a timestamped audit event.
5) Authorized staff review the request and update its status with remarks.
6) The citizen checks progress using authenticated access or a safe reference-based mechanism.
7) Aggregated, non-personal service statistics are shown on the public dashboard.

VI. METHODOLOGY AND EVALUATION PLAN

This paper uses a source-code review and literature-based design analysis. The supplied website project archive was inspected to identify pages, API routes, data storage patterns, and apparent security controls. Four supplied IEEE papers were reviewed for themes relevant to governance transparency, AI, citizen feedback, and digital inclusion. The review did not include a formal usability study, a penetration test, a production database, or measured performance experiments.

The next implementation phase should evaluate the system using the following test plan:

- Functional tests: registration, login, application submission, status changes, contact submissions, validation errors, and unauthorized requests.
- Security tests: password hashing, access-control checks, session handling, rate limiting, CORS configuration, injection resistance, and sensitive-data exposure.
- Usability tests: task completion by residents with different levels of digital literacy, including Kannada-speaking users and mobile-only users.
- Reliability tests: concurrent application submissions, backup restoration, server restart behaviour, and database consistency.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 10 , October - 2026

- Service-quality measures: application completion rate, time to submit a request, percentage of requests with status updates, and median resolution time.

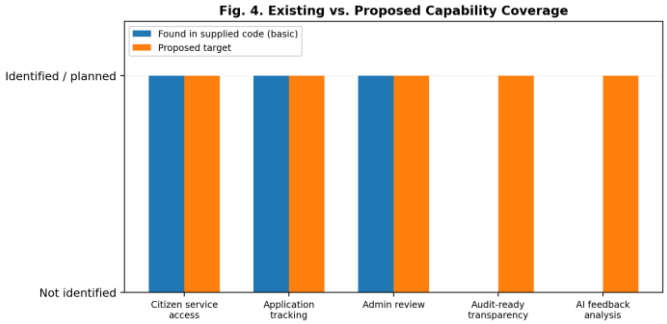


Fig. 3. Capability coverage: code-review observations and proposed targets; not measured performance.

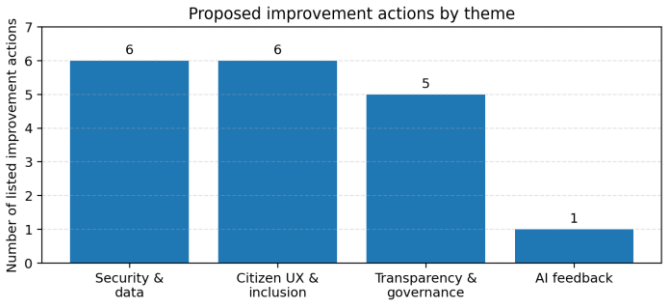


Fig. 4. Count of proposed improvement actions by theme, based on the recommendations listed in Section IV; counts are not performance measurements.

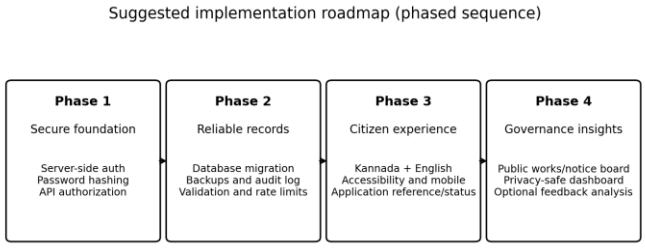


Fig. 5. Suggested phased implementation roadmap; sequence is a planning proposal, not a measured schedule.

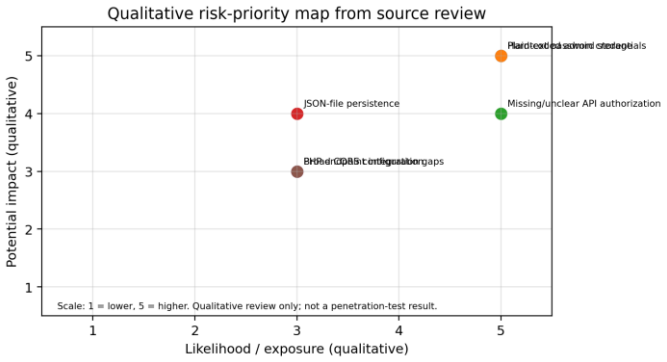


Fig. 6. Qualitative risk-priority map derived from source-code review; ratings should be validated through formal security testing.

No numerical outcomes are reported in this draft because the required test runs and participant data have not been provided. Any future results section should report the test environment, sample size, procedure, observed values, and limitations rather than estimated or invented figures.

VII. DISCUSSION

The current project provides a useful starting point because it brings public service information and basic application workflows into one web interface. Its clearest near-term value is not advanced AI or blockchain, but making the existing service-request process dependable, secure, and understandable. The code review indicates that security and persistence should be addressed before adding more complex features. A database-backed, role-protected workflow would also create a stronger foundation for transparent public reporting and later analytics.

The proposed improvements are aligned with the reviewed literature but are deliberately phased. Secure authentication, reliable data storage, accessibility, bilingual content, and clear application tracking should be treated as core requirements. Public-works transparency and text-based feedback analysis can follow once data quality and governance processes are established. Blockchain should remain an optional future study, not a claim of the present implementation.

VIII. LIMITATIONS

The analysis is limited to the supplied project archive, the visible response from the deployed website at the time of review, four user-provided IEEE papers, and a public description of eGramSwaraj. It does not establish that all listed pages or APIs are operational in the live deployment. The paper does not claim official government affiliation, formal integration with eGramSwaraj, deployment of blockchain, use of AI models, or measurable improvements in citizen outcomes. Those claims would require implementation evidence and testing.

IX. CONCLUSION AND FUTURE SCOPE

SmartMandya is a web-based Gram Panchayat project with public information pages, citizen application endpoints, basic application-status handling, administrative views, and contact-form processing. The analysis identifies a practical improvement path: secure the authentication and API layer, replace file-based persistence with a managed database, improve Kannada-English accessibility, provide clear application tracking, and establish audit-ready records. Future work may evaluate public-works dashboards and privacy-conscious text feedback analysis. Any AI or blockchain extension should be justified by a defined governance need and validated through transparent testing. The next milestone is to implement the security and database changes, then evaluate usability, reliability, and service outcomes with documented evidence.

REFERENCES

[1] A. Ajay, S. M. R. Shrihari, S. K. Suchitra, U. B. S., V. Nandini, and V. S. R. Vandana, “Development of E-Governance Services in Smart Cities using Artificial Intelligence and Blockchain,” 2024 International Conference on Knowledge Engineering and Communication Systems (ICKECS), 2024, doi: 10.1109/ICKECS61492.2024.10616607.

[2] A. Mainali, P. Devkota, N. Thapaliya, P. Shrestha, and S. Yadav, “Blockchain-based E-Governance for Transparent Allocation and Utilization of Local Government Funds in Rural Regions,” Proceedings of the 9th International Conference on Inventive Systems and Control (ICISC), 2025, doi: 10.1109/ICISC65841.2025.11188433.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 10 , October - 2026

[3] C. Nagesh, B. Divyasree, K. Madhu, T. Allisha, S. Datta Koushik, and P. Naresh, "Enhancing E-Government through Sentiment Analysis: A Dual Approach Using Text and Facial Expression Recognition," 2024 International Conference on Science Technology Engineering and Management (ICSTEM), 2024, doi: 10.1109/ICSTEM61137.2024.10560678.

[4] T. Zheliuk, D. Shushpanov, A. Zhukovska, O. Brechko, V. Ostroverkhov, and V. Posheliuzhnyu, "Information Technologies of Public Governance in the Context of Digitalization and Demographic Challenges," 15th International Conference on Advanced Computer Information Technologies (ACIT), 2025, doi: 10.1109/ACIT65614.2025.11185850.

[5] National Informatics Centre, Government of India, "eGramSwaraj," project overview. Available: https://www.nic.gov.in/project/egramswaraj/ (accessed 17 Oct. 2026).

(This work is licensed under a Creative Commons Attribution 4.0 International License.)