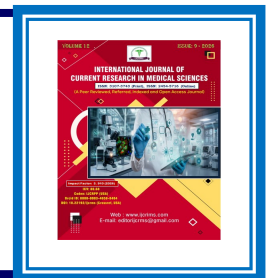




International Journal of Current Research in Medical Sciences

ISSN: 3107-3743 (Print), ISSN: 2454-5716 (Online)
(A Peer Reviewed, Indexed and Open Access Journal)
www.ijcrims.com



Original Research Article

Volume 12, Issue 9 -2026

DOI: <http://dx.doi.org/10.22192/ijcrms.2026.12.09.005>

Development of an AI-driven epidemic surveillance and pandemic prevention system

Lovingkindness Audu¹, Hussein Abdulsalam Hassan²

Department of Software Engineering, Faculty of Computing and Informatics,
Confluence University of Science and Technology, Osara, Nigeria.

Email: audule@custech.edu.ng/hassanhuss1027@gmail.com

Corresponding Author: Lovingkindness Audu- audule@custech.edu.ng

Abstract

Nigeria's conventional disease surveillance processes depend heavily on structured clinical reporting and hierarchical notification, which can introduce delays between the appearance of an outbreak signal and official awareness. This study presents the design and implementation of an AI-driven epidemic surveillance and pandemic prevention system tailored to the Nigerian public-health context. The proposed system combines automated web scraping, Natural Language Processing (NLP), weighted risk scoring, real-time NoSQL synchronization, geospatial visualization, public symptom reporting, and an AI-assisted chatbot. Nigerian online news and health sources are collected by a Python-based scraper and transformed into structured surveillance reports. An NLP risk engine filters irrelevant phrases, identifies disease and symptom indicators, assigns a risk score from 0 to 100, and classifies reports as Low, Medium, or High risk. High-risk information is synchronized to Firebase Cloud Firestore and displayed through an interactive Leaflet.js dashboard. The public reporting interface provides a second stream of signals, while a Gemini 2.5-powered chatbot supplies preliminary safety guidance. Functional testing covered scraping, chatbot integration, risk scoring, real-time dashboard updates, and anonymous public reporting. The results demonstrate software feasibility for localized, event-based surveillance; however, the system remains a surveillance aid rather than a clinical diagnostic instrument.

Keywords: Epidemic Surveillance, Event-Based Surveillance, Natural Language Processing, Web Scraping, Risk Scoring, Geospatial Dashboard, Firebase, Digital Epidemiology, Nigeria.

1. Introduction

The rapid movement of people, information and goods across borders has made infectious-disease events increasingly difficult to contain when detection is delayed [1]. The COVID-19 pandemic demonstrated that surveillance systems must provide actionable information early enough

for public-health authorities to intervene. In Nigeria, recurring threats including Lassa fever, cholera and meningitis coexist with the possibility of newly introduced pathogens [2]. The project therefore approaches epidemic surveillance as a software-engineering problem in which digital signals can complement established clinical reporting.

Traditional Indicator-Based Surveillance (IBS) follows a structured reporting chain: a case is identified at a health facility, documented, forwarded through Local Government and State structures, and ultimately reported to the national level [3]. This structure is valuable because clinical data can be verified, but the project identifies reporting latency as a major weakness, particularly where connectivity, staffing and paper-based processes create additional delays. Event-Based Surveillance (EBS) offers a complementary approach by actively searching for signals in news reports, community information and other unstructured sources [4].

Digital epidemiology extends this idea by treating online information as a potential early-warning stream. News headlines, articles, public submissions and other digital traces can contain symptoms, disease names, locations and outbreak terminology before a formal case report is available [5]. Automated extraction and NLP can reduce the manual effort required to inspect these sources continuously [6]. The system developed in this study is designed around that principle and is localized to Nigerian online sources.

1.1 Statement of the Problem

Three related gaps motivate the system. First, manual reporting can create a time gap between an emerging event and official awareness [7]. Second, conventional structured surveillance platforms are not designed to continuously interpret unstructured online text, leaving potentially useful digital signals outside the main surveillance workflow [8]. Third, decision-makers need a spatial view of where signals are clustering; static reports and spreadsheets do not provide the same immediate visual context as a live geospatial dashboard [9].

1.2 Aim and Objectives

The aim is to design and implement a web-based AI-driven epidemic surveillance system that gathers online disease signals, analyzes them with NLP, maps emerging threats, and accepts direct public symptom reports.

1. Develop an autonomous Python web scraper for Nigerian news websites, health blogs and relevant feeds.
2. Design an NLP-based risk-analysis engine that filters noise and calculates a risk score for each report.
3. Create a real-time geospatial dashboard that displays high-risk reports and emerging clusters.
4. Implement a public symptom-reporting interface that provides an additional surveillance signal.

1.3 Scope and Significance

The implementation is software-focused. It covers Python-based extraction and text processing, Firebase Cloud Firestore for real-time data storage, and a browser-based dashboard using HTML5, CSS3, JavaScript and Leaflet.js [10]. The target data environment is Nigeria, including national and local news sources and Nigerian geographic locations. No hardware sensors, laboratory experiments or biological sample collection are included. The system is intended to support surveillance prioritization and early warning, not to replace clinical diagnosis or laboratory confirmation [11].

The principal significance is the combination of three complementary streams: automated digital-source monitoring, public reporting, and geospatial presentation [12]. Together they provide a practical framework for moving from purely reactive reporting toward earlier identification of signals that deserve human review [13].

2. Review of related literature

Disease surveillance has evolved from predominantly structured case reporting toward hybrid approaches that combine clinical data with digital and event-based signals [14]. SORMAS demonstrates the value of digitizing outbreak management and contact tracing, but the project literature review notes that systems centered on clinical workflows remain dependent on a case reaching a health facility. Event-based approaches instead seek signals from reports, rumours and other unstructured sources [15].

HealthMap established an important precedent for automated classification and visualization of Internet media reports. Web-based disease detection research similarly demonstrates that digital sources can complement conventional surveillance [16]. However, the project identifies a localization gap: a Nigerian implementation must account for local media sources, Nigerian place names, limited-resource connectivity and language variation. The literature also highlights ethical and governance issues around harvesting digital health information [17].

Prior work on web scraping shows that Python-based extraction can make otherwise difficult-to-access outbreak information machine-readable. NLP and machine learning provide methods for transforming text into structured indicators, while real-time NoSQL systems can synchronize updates to multiple clients [18]. Geospatial dashboards then provide a decision-support layer for interpreting the location and clustering of detected signals [19].

2.1 Research Gap

Existing approach	Strength	Gap identified by the project
SORMAS	Digital case registration, contact tracing and outbreak management.	Primarily structured around health-facility workflows; does not continuously harvest public web signals.
HealthMap	Automated Internet-media monitoring and visualization.	Global orientation requires adaptation for Nigerian local sources and context.
ProMED-mail	Rapid expert-reviewed infectious-disease alerts.	Human review can become a bottleneck when signal volume is high.
Proposed system	Combines web scraping, NLP scoring, public reporting and real-time mapping.	Requires internet access, careful validation and further multilingual/clinical evaluation.

Table 1: Comparison of surveillance approaches discussed in the project.

3. Methodology and System design

The project uses the agile development methodology, which is very reliable for modern day software development, and it ensures that the

entire project is broken down into sprints. This iterative approach is ideal for adapting to changing requirements and continuous feedback [20].



Figure 1: Agile Development methodology

The design separates presentation, application logic and data management so that data collection, risk analysis and visualization can evolve independently. The resulting architecture consists of an HTML5/CSS3/JavaScript presentation layer, a Python application layer, and a Firebase Cloud Firestore data layer.

3.1 System Architecture

The application begins with news or social-media-derived web content. The Python scraper

retrieves raw HTML or RSS content, cleans the text and forwards the resulting report to the NLP processor. The risk-analysis stage identifies disease, symptom, outbreak and severity indicators and assigns a score. High-risk information is written to Firestore, while the dashboard subscribes to data changes and updates the map and surveillance ticker.

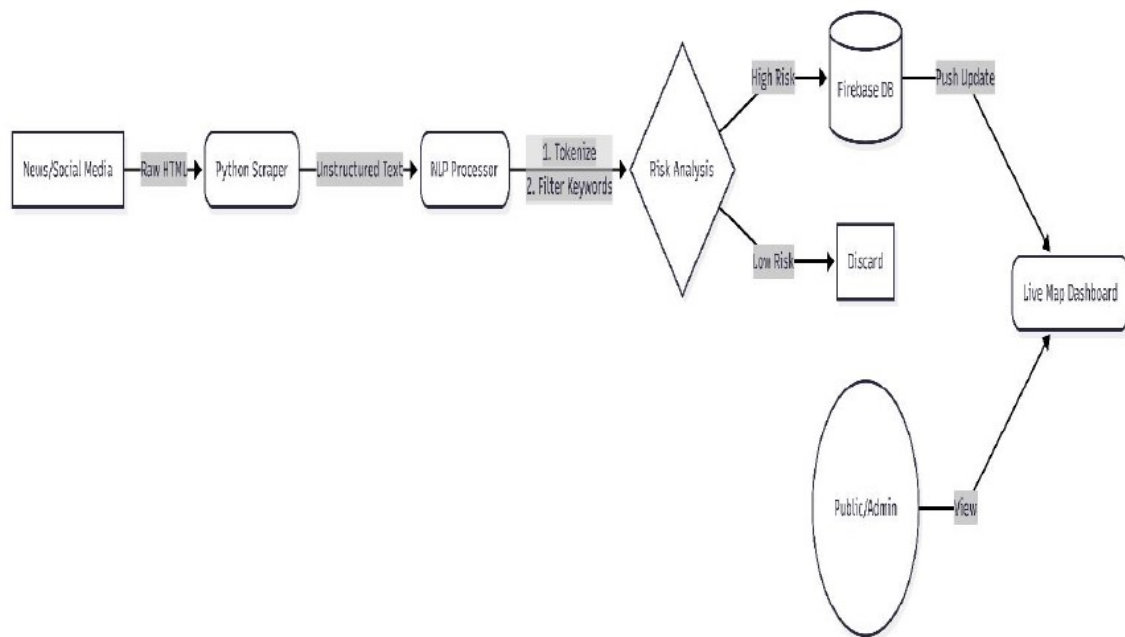


Figure 2: Data-flow diagram showing transformation of online content into scored and mapped surveillance information.

3.2 Data Model and Real-Time Storage

The Firestore design uses three principal collections: reports, locations, and users. The reports collection stores scraped or submitted surveillance items, including title, URL, risk score, location and timestamp. The locations collection provides geographic reference data for Nigerian states and the Federal Capital Territory. The users collection supports administrative access. Firestore was selected because its real-time synchronization model fits a dashboard in which newly detected signals should appear without requiring a full-page refresh [21].

3.3 NLP Risk-Scoring Engine

The implemented risk engine uses weighted keyword and phrase matching. A report begins with a base score of 10. Symptom indicators contribute up to 30 points, outbreak terms contribute up to 30 points, disease mentions contribute 25 points, and numeric severity indicators contribute 20 points. Additional rules recognize specific diseases and combinations such as cholera with outbreak/case language, vomiting with diarrhea, and meningitis with stiff neck. The final score is constrained to 0-100.

Signal	Contribution	Purpose
Base score	10	Initial signal value.
Symptoms	Up to +30	Raises risk when multiple compatible symptoms appear.
Outbreak terms	Up to +30	Captures outbreak/epidemic/confirmed/spreading language.
Disease mention	+25	Prioritizes explicit disease references.
Numeric severity	+20	Uses reported counts/severity as an additional signal.
Specific disease/context rules	+10 to +20	Strengthens combinations considered epidemiologically salient.

Table 2: Main rule-based components of the implemented risk engine.

Risk levels are classified as High for scores of 75 or above, Medium for scores from 40 to 74, and Low below 40. The implementation also supports an optional machine-learning model; where available, the model may override the rule-based result when its confidence exceeds 0.6. The engine includes explicit noise phrases such as "election fever" and "football fever" to reduce false positives caused by non-medical uses of health-related words.

3.4 Location Extraction and Network Resilience

Location extraction uses a Nigerian location dictionary containing states, the Federal Capital Territory, and selected cities. The first matching location is used as the primary location when appropriate, with a fallback location available to the processing pipeline. Because the scraper and chatbot depend on external network services, the implementation also includes retry logic using exponential backoff.

4. System implementation

Implementation combines autonomous extraction, text analysis, cloud storage, geospatial visualization and public interaction into a single browser-accessible application. Python is the primary backend runtime, BeautifulSoup4 and RSS parsing support extraction, NLTK supports text processing, Firebase Admin provides database access, and Leaflet.js provides the map interface.

4.1 Autonomous Web Scraper

The scraper supports HTML and RSS sources. For HTML sources it identifies article links, cleans titles and summaries, resolves relative URLs, and prevents duplicate URLs within a retrieval cycle. For RSS sources it parses feed entries and creates normalized report objects containing title, summary, URL, source name, source type and publication information.

4.2 AI-Assisted Chatbot

The server-side /api/chat route integrates the Gemini 2.5 large language model to provide conversational symptom guidance. The chatbot is positioned as an information and safety-guidance component rather than a diagnostic authority. Its role is to provide immediate preliminary guidance while encouraging appropriate medical attention when symptoms or exposure concerns warrant it.

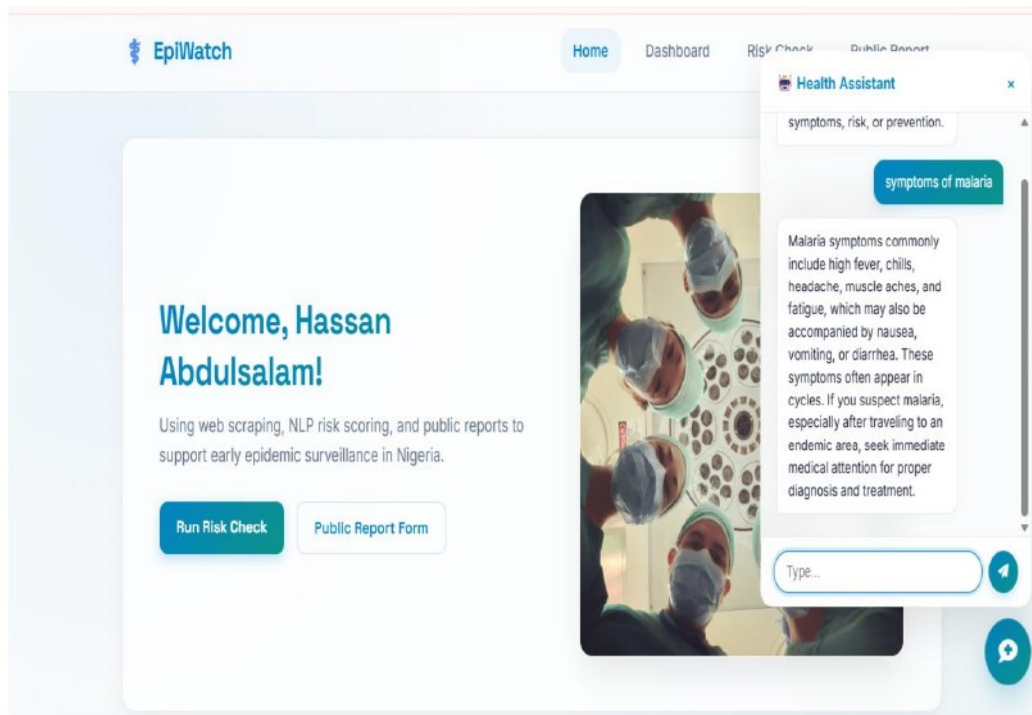


Figure 3: AI-assisted health interface providing preliminary symptom guidance and links to surveillance functions.

4.3 Geospatial Dashboard

The administrative command center combines a Leaflet-based map with a live surveillance ticker. Firestore updates are reflected in the interface so

that new high-risk records can be visualized without a manual page reload. This presentation layer is intended to help surveillance personnel recognize spatial clustering and prioritize further investigation.

Figure 4: Administrative dashboard integrating a Leaflet geospatial view and live surveillance ticker.

4.4 Public Symptom Reporting

The public reporting interface provides an additional source of surveillance information. A user can submit symptoms from a Local

Government Area without requiring the report to pass through a paper-based workflow. The report is written directly to Firestore, where it can become part of the broader surveillance stream.

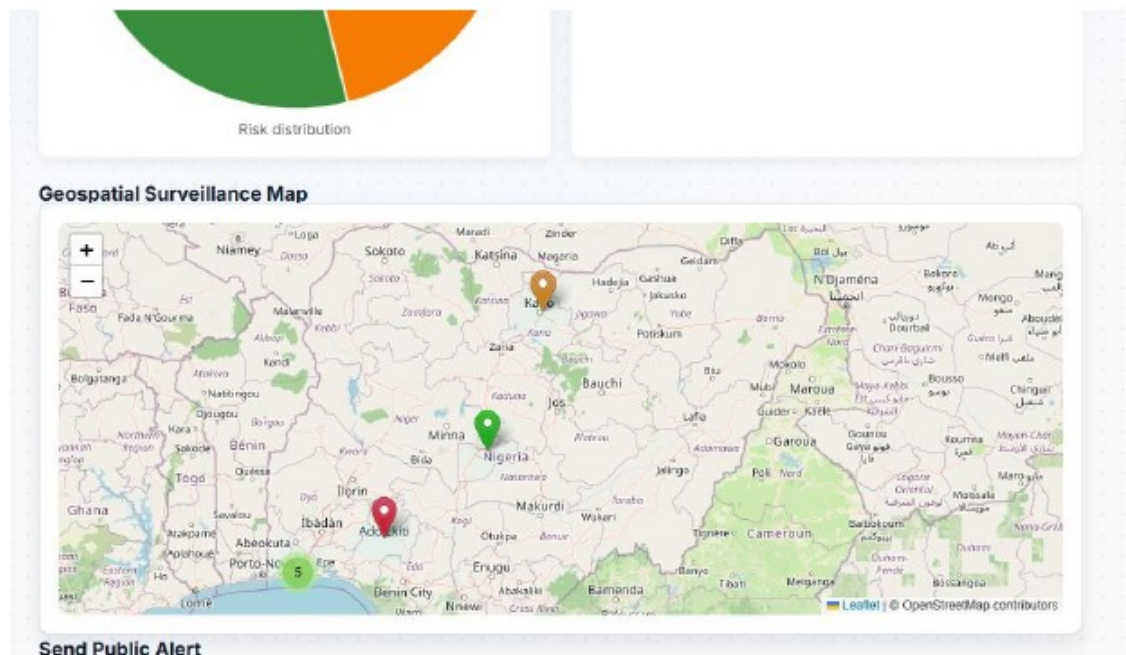


Figure 5: Public symptom-reporting interface designed for direct submission to the surveillance database.

5. Testing and Results

Testing was performed across unit, integration, system and user-acceptance levels. The objective

was to verify that the major modules worked together and that the implementation addressed the practical gaps identified in the problem statement.

ID	Test type	Module/input	Status
TC-01	Unit	scraper.py with a Punch News URL	Pass
TC-02	Integration	/api/chat with simulated Lassa fever symptoms	Pass
TC-03	System	risk_engine.py with 'Cholera' and 'Death'	Pass
TC-04	System	New high-risk Firestore location	Pass
TC-05	UAT	Anonymous public symptom report from a simulated rural LGA	Pass

Table 3: Summary of the system unit, integration, system and acceptance tests.

5.1 Findings

- Reduced manual-reporting dependency: the public reporting form demonstrated direct submission into Firestore.
- Digital-source processing: the scraper and NLP pipeline successfully transformed online content into structured surveillance records.
- Automated prioritization: the weighted risk engine produced a high-risk score of 88 for the synthetic 'Cholera' and 'Death' test case.
- Real-time visualization: a new high-risk location propagated to the Leaflet dashboard without a manual reload.
- AI-assisted guidance: the chatbot successfully integrated with the backend and returned a safety-oriented response to the simulated Lassa fever query.

6. Discussion

The implemented system demonstrates that event-based surveillance can be expressed as a practical software pipeline: collect, clean, score, store, visualize and review. The approach does not attempt to replace Nigeria's existing surveillance infrastructure. Instead, it provides a complementary early-warning layer that can surface signals from sources that are not naturally represented as structured clinical records.

A key design choice is the use of explainable weighted rules before optional machine-learning intervention. A surveillance officer can inspect why a report received a high score because the major contributors correspond to identifiable terms and combinations. This is particularly useful in an early prototype where labeled Nigerian epidemic text may be limited. The optional ML override allows the architecture to evolve toward learned models without discarding the transparent baseline.

The real-time Firestore and Leaflet combination addresses a separate operational issue: even a useful alert has limited value if decision-makers cannot see where it occurred. Mapping state or city matches gives the system a spatial layer that can support triage and further investigation.

Nevertheless, a mapped signal remains a signal; it must be verified before public-health action is based on it.

The public symptom interface also changes the direction of information flow. Instead of waiting for information to travel upward through formal channels, the platform allows a citizen-level signal to enter the digital system directly. This may be particularly useful in locations where formal reporting is slow, but it also introduces the need for strong validation, privacy controls, abuse prevention and human review.

6.1 Ethical and Operational Considerations

Digital epidemiology raises privacy, consent, data-quality and governance concerns. The project therefore treats the system as a surveillance aid rather than an autonomous decision-maker. Public submissions should be minimized to the information necessary for surveillance, administrative access should be controlled, and alerts should be verified by qualified personnel. The use of web sources should also respect source policies and applicable legal and ethical requirements.

7. Limitations

- The system is not a clinical diagnostic tool and cannot establish laboratory-confirmed disease.
- Continuous operation depends on Internet access to external websites, APIs and cloud services.
- The current NLP implementation is primarily oriented toward English and Pidgin English and may perform poorly on pure Hausa, Igbo or Yoruba reports.
- Web-source structure can change, requiring scraper maintenance when page layouts or feeds change.
- The project demonstrates functional software tests rather than large-scale epidemiological validation against a longitudinal gold-standard dataset.

8. Conclusion and Recommendations

This study presents a localized AI-driven epidemic surveillance system that combines automated web scraping, NLP risk scoring, real-time cloud storage, geospatial visualization, public reporting and AI-assisted guidance. The implemented modules passed the project's functional test cases, demonstrating that the proposed architecture can operate as an integrated software prototype. Its main contribution is a practical framework for complementing structured surveillance with digital event signals relevant to the Nigerian context.

The results should be interpreted as evidence of software feasibility rather than proof of clinical or population-level accuracy. Deployment in a real public-health environment would require validation against verified outbreak datasets, formal governance, security and privacy controls, monitoring for false positives and false negatives, and sustained collaboration with public-health professionals.

8.1 Recommendations

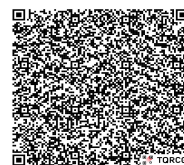
1. Consider controlled integration of AI-assisted triage and information tools into official health portals, with clear human-oversight boundaries.
2. Evaluate deployment to Local Government Area disease-notification officers as a complementary visualization and alerting layer.
3. Expand source coverage while maintaining source-policy compliance and quality controls.
4. Build multilingual NLP support for Hausa, Yoruba and Igbo.
5. Investigate offline or store-and-forward synchronization for public reporting in areas with intermittent connectivity.
6. Develop larger, expert-annotated Nigerian outbreak datasets for rigorous model evaluation.

References

1. Achieng, M. S., & Ogundaini, O. O. (2024). Big Data Analytics for Integrated Infectious Disease Surveillance in sub-Saharan Africa. *South African Journal of Information Management*, 26(1), 1668. <https://doi.org/10.4102/sajim.v26i1.1668>.
2. Piccialli, F., di Cola, V., Giampaolo, F. et al. The Role of Artificial Intelligence in Fighting the COVID-19 Pandemic. *Inf Syst Front* 23, 1467–1497 (2021). <https://doi.org/10.1007/s10796-021-10131-x>
3. Parums D.V. (2023). Infectious Disease Surveillance Using Artificial Intelligence (AI) and its Role in Epidemic and Pandemic Preparedness. *Editorial Med Sci Monit*.29:e941209. <https://doi.org/10.12659/MSM.941209>.
4. Kibibi Muthoni L. (2024). The Role of AI in Enhancing Public Health Surveillance. *Eurasian Experiment Journal of Biological Sciences*, 5(2):37-40. https://kiu.ac.ug/assets/publications/2686_the-role-of-ai-in-enhancing-public-health-surveillance.pdf.
5. Rahman, M.S., Safa, N.T., Sultana, S., Salam, S., Karamelic-Muratovic, A. & Overgaard, H.J. (2022). Role of artificial intelligence internet of things (AI-IoT) based emerging technologies in the public health response to infectious diseases in Bangladesh. *Parasite Epidemiology and Control*, 18(e00266). <https://doi.org/10.1016/j.parepi.2022.e00266>.
6. Popescu, S.V. (2023). Using AI as a Tool for Infectious Disease Surveillance. <https://www.contagionlive.com/view/using-ai-as-a-tool-for-infectious-disease-surveillance>.
7. Aldubikhi, A.I.S. (2022). The Role of Artificial Intelligence in the Control and Prevention of Infectious Diseases. *Journal of Health Informatics in Developing Countries* 1(7). <https://www.jhids.org/index.php/jhids/article/view/387/358>.

8. Mokhtari, A. Artificial Intelligence for Public Health Surveillance: Harnessing new advances in predictive modeling. <https://www.boozallen.com/insights/ai-research/artificial-intelligence-for-public-health-surveillance.html>
9. Ershad, Mr & Ali, Ershad. (2020). Geographic Information System (GIS): Definition, Development, Applications & Components.
10. Aranha, Diego & Dominguez Perez, Luis & Mrabet, Amine & Schwabe, Peter. (2016). Software Implementation. 10.1201/9781315370170-11.
11. Anjaria, P., Asediya, V., Bhavsar, P., Pathak, A., Desai, D., & Patil, V. (2023). Artificial Intelligence in Public Health: Revolutionizing Epidemiological Surveillance for Pandemic Preparedness and Equitable Vaccine Access. *Vaccines*, 11(7), 1154. <https://doi.org/10.3390/vaccines11071154>.
12. Tarek, M., Roche, S., & Berthold, E. (2026). Using Geospatial Data to Monitor and Evaluate Local SDG Implementation: A Systematic Review. *Sustainability*, 18(16), 8139. <https://doi.org/10.3390/su18168139>
13. Madon, Temina & Agnihotri, Anustubh & Gadgil, Ashok. (2022). A Practical Framework for Research. 10.1007/978-3-030-86065-3_3.
14. Tanui, C. K., Ndembi, N., Kebede, Y. & Tessema, S. K. (2024). Artificial intelligence to transform public health in Africa. *The Lancet Infectious Diseases* e542, 24-9. [https://doi.org/10.1016/S1473-3099\(24\)00435-3](https://doi.org/10.1016/S1473-3099(24)00435-3).
15. Syrowatka, A., Kuznetsova, M., Alsubai, A. et al. (2021). Leveraging artificial intelligence for pandemic preparedness and response: a scoping review to identify key use cases. *npj Digit. Med.* 4, 96. <https://doi.org/10.1038/s41746-021-00459-8>.
16. Rajora, Harish & Pun, Narinder & Sonbhadra, Sanjay & Agarwal, Sonali. (2021). Web based disease prediction and recommender system. 10.48550/arXiv.2106.02813.
17. Abbood, A., Ullrich, A., Busche, R., & Ghazzi, S. (2020). EventEpi—A natural language processing framework for event-based surveillance. *PLOS Computational Biology*, 16(11), e1008277. <https://doi.org/10.1371/journal.pcbi.1008277>
18. Al-Garadi, M. A., Yang, Y.-C., & Sarker, A. (2022). The Role of Natural Language Processing during the COVID-19 Pandemic: Health Applications, Opportunities, and Challenges. *Healthcare*, 10(11), 2270. <https://doi.org/10.3390/healthcare10112270>
19. Shen, C., Chen, A., Luo, C., Zhang, J., Feng, B., & Liao, W. (2020). Using Reports of Symptoms and Diagnoses on Social Media to Predict COVID-19 Case Counts in Mainland China: Observational Infoveillance Study. *Journal of Medical Internet Research*, 22(5), e19421. <https://doi.org/10.2196/19421>
20. Omonije, Ajibola. (2024). Agile Methodology: A Comprehensive Impact on Modern Business Operations. *International Journal of Science and Research (IJSR)*. 13. 132-138. 10.21275/SR24130104148.
21. Napoli, Marco. (2019). Adding the Firebase and Firestore Backend. 10.1002/9781119550860.ch14.

Access this Article in Online



Website:

www.ijcrims.com

Subject:

Health Informatics

Quick Response Code

DOI: [10.22192/ijcrms.2026.12.09.005](https://doi.org/10.22192/ijcrms.2026.12.09.005)

How to cite this article:

Lovingkindness Audu, Hussein Abdulsalam Hassan. (2026). Development of an AI-driven epidemic surveillance and pandemic prevention system. *Int. J. Curr. Res. Med. Sci.* 12(9): 46-55.
DOI: <http://dx.doi.org/10.22192/ijcrms.2026.12.09.005>