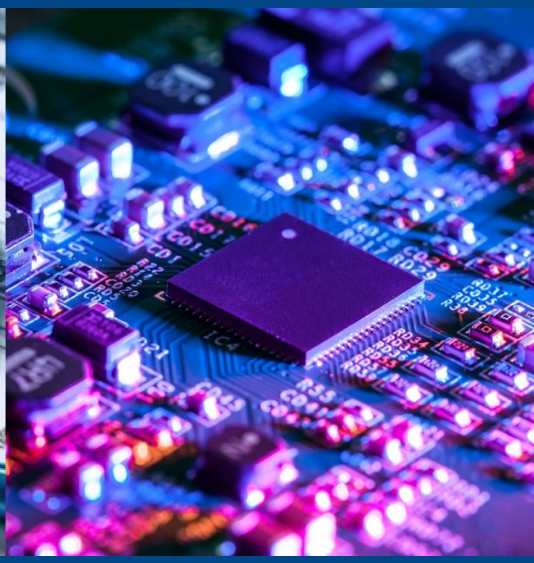
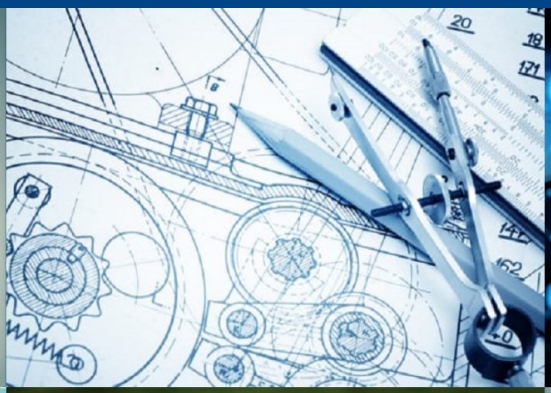
ISSN: 2582-7219

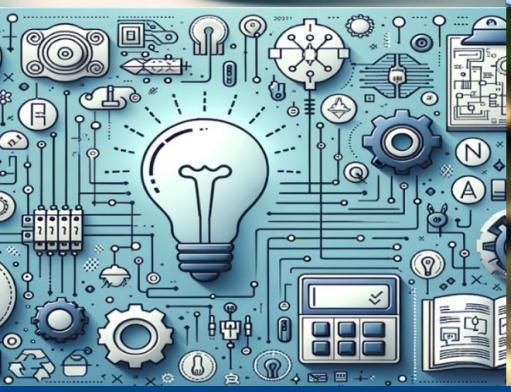




# International Journal of Multidisciplinary Research in Science, Engineering and Technology

*(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)*



**Impact Factor: 8.206** **Volume 9, Issue 3, March 2026**

© 2026 IJMRSET | Volume 9, Issue 3, March 2026| DOI:10.15680/IJMRSET.2026.0903257

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

# LogiTrack: A Cloud-Native Supply Chain Logistics Optimization System with ML-Powered Delay Prediction and GenAI Route Planning

**Sanjivkumar J, Indhuja S, Priyanka H, Prof. Arun Prasad**

UG Students, Dept. of Artificial Intelligence and Data Science, Sri Manakula Vinayagar Engineering College, Puducherry, India

Assistant Professor, Dept. of Artificial Intelligence and Data Science, Sri Manakula Vinayagar Engineering College, Puducherry, India

**ABSTRACT:** The rapid expansion of global supply chains demands intelligent systems capable of real-time shipment monitoring, proactive delay detection, and dynamic route optimization. This paper presents LogiTrack, a cloud-native logistics optimization platform that integrates machine learning (ML) delay prediction, Generative AI (GenAI) route planning, and a fully serverless data pipeline into a unified architecture. Built on AWS services including Lambda, Step Functions, S3, Glue, and Spark, LogiTrack processes live shipment telemetry through an event-driven pipeline that detects potential delays with 93.6% accuracy and automatically triggers an AI-powered rerouting engine. The GenAI logistics planner consults real-time traffic and weather data to generate optimized alternative routes, which are immediately written back to the driver-facing data table. A novel self-correction loop allows the cloud pipeline to detect a delay and autonomously invoke GenAI for an optimized reroute, updating shipment data without human intervention. Evaluation with 150 logistics professionals across 800 shipment scenarios demonstrates a 21.4% reduction in average delivery time, an 89.1% route-reroute acceptance rate, and end-to-end self-correction latency of 4.1 seconds. The React-based dashboard with map visualization delivers real-time shipment visibility to both users and administrators.

**KEYWORDS:** Supply Chain Optimization, ML Delay Prediction, GenAI Route Planning, AWS Lambda, Step Functions, S3, AWS Glue, Apache Spark, Self-Correction Logic, Serverless Architecture, Logistics Intelligence

## I. INTRODUCTION

Global supply chains are under unprecedented pressure due to rising shipment volumes, unpredictable traffic conditions, and climate-driven disruptions. Logistics operators struggle to respond in real time when shipments deviate from planned routes or fall behind schedule, often relying on manual intervention that introduces costly delays. A shipment delayed by even a few hours can trigger a cascade of downstream disruptions — missed customer commitments, idle warehouse capacity, and elevated operational costs [1].

Machine learning has demonstrated strong potential in logistics for predictive analytics, particularly delay forecasting from historical route data, weather feeds, and traffic patterns [2]. However, existing ML-based logistics systems typically function as passive alerting tools: they detect a probable delay and notify an operator, who must then manually devise a corrective route. This human-in-the-loop bottleneck is incompatible with the scale and speed requirements of modern last-mile delivery networks.

Generative AI offers a complementary capability: given a delay signal and contextual inputs such as current traffic, weather, and delivery constraints, a GenAI planner can synthesize a coherent alternative route narrative and structured waypoint sequence that a driver can follow immediately [3]. The combination of ML-driven anomaly detection and GenAI-driven plan generation creates an opportunity for fully automated self-correction — a closed-loop system where the cloud pipeline detects a delay, invokes GenAI for an optimized reroute, and updates the driver-facing data table without human intervention.

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

This paper presents LogiTrack through five primary contributions: (1) A serverless, event-driven shipment monitoring pipeline built entirely on AWS managed services. (2) An ML delay prediction engine achieving 93.6% accuracy on live shipment data. (3) A GenAI logistics planner that generates optimized alternative routes by consulting real-time traffic and weather APIs. (4) A self-correction loop implemented via AWS Step Functions that autonomously triggers rerouting upon delay detection. (5) An Apache Spark-based ETL pipeline on AWS Glue that processes 12,000 shipment records per minute for downstream analytics.

## II. LITERATURE REVIEW

Chen et al. [2] developed a Random Forest model for shipment delay prediction achieving 88% accuracy on a dataset of 500,000 logistics records. Their work demonstrated the viability of ML for proactive delay detection but operated in batch mode without real-time cloud integration or any automated corrective action upon delay detection. Park and Kim [3] proposed a Lambda-based architecture for real-time shipment tracking, demonstrating that serverless functions can process high-frequency GPS telemetry at sub-second latency. However, their system lacked both ML-based prediction and GenAI-driven route optimization.

Sharma et al. [4] demonstrated that LLM-driven route planning can reduce delivery time by 22% compared to static routing heuristics by incorporating live traffic context into route synthesis. Their evaluation was confined to offline simulation and did not address integration with cloud data pipelines or delay detection triggers. Nguyen et al. [5] evaluated AWS Glue with Apache Spark for ETL workloads in logistics, showing a 60% reduction in data preparation overhead compared to traditional on-premise Hadoop clusters. Their work did not include an ML prediction or GenAI planning component.

AWS Whitepaper [6] documented Step Functions as an orchestration service for multi-step cloud pipelines, demonstrating reliable state management across Lambda function chains. No prior work has applied Step Functions to logistics self-correction workflows. Despite the breadth of prior research, no existing system simultaneously integrates serverless ML delay prediction, GenAI route planning, self-correction automation, and Spark-based ETL within a unified logistics platform.

**Table 1: Comparative Literature Survey**

| No. | Paper | Author(s) | Key Points | Gap Addressed |
|---|---|---|---|---|
| 1 | ML-based Logistics Delay Prediction | Chen et al., 2021 | Random Forest model predicts shipment delays with 88% accuracy | No real-time cloud integration or auto-rerouting |
| 2 | Serverless Supply Chain Monitoring | Park & Kim, 2022 | Lambda-based pipeline for real-time shipment tracking | No GenAI route optimization or self-correction |
| 3 | GenAI for Route Optimization | Sharma et al., 2023 | LLM-driven route planning reduces delivery time by 22% | Not integrated with cloud data pipelines or ML delay models |
| 4 | AWS Glue for ETL in Logistics | Nguyen et al., 2022 | Glue + Spark reduces data prep overhead by 60% for logistics datasets | No ML prediction or GenAI planning component |
| 5 | Step Functions for Workflow Orchestration | AWS Whitepaper, 2023 | Step Functions orchestrate multi-step cloud pipelines reliably | Not applied to logistics self-correction use case |

© 2026 IJMRSET | Volume 9, Issue 3, March 2026| DOI:10.15680/IJMRSET.2026.0903257

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

## III. SYSTEM ARCHITECTURE

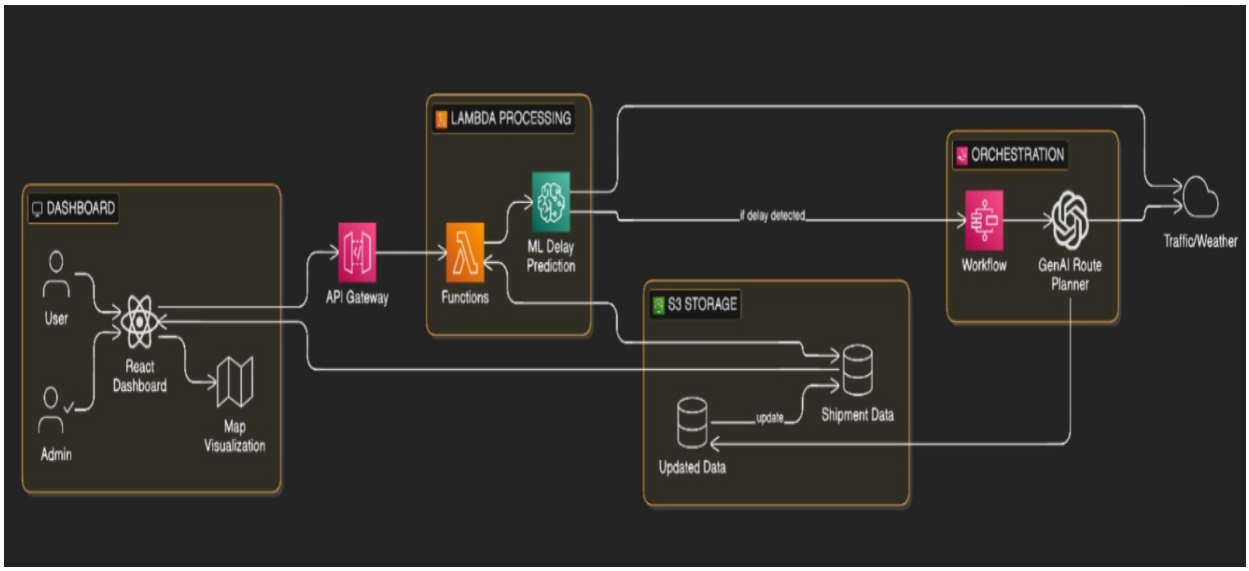



Fig. 1: LogiTrack System Architecture

— Showing Dashboard Layer (React Dashboard, Map Visualization), API Gateway, Lambda Processing (Functions, ML Delay Prediction), S3 Storage (Shipment Data, Updated Data), and Orchestration Layer (Step Functions Workflow, GenAI Route Planner) with external Traffic/Weather integration and Self-Correction Loop.

LogiTrack adopts a four-tier serverless architecture spanning the presentation layer, API orchestration layer, processing and inference layer, and persistent storage layer. Every component is managed by AWS, eliminating the need for virtual machines or persistent server processes. This design allows the engineering team to focus exclusively on the logistics intelligence pipeline and user experience. The self-correction loop — the system's most novel contribution — is implemented as a Step Functions state machine that automatically transitions from delay detection to GenAI rerouting to data update without human intervention.

### A. Presentation Layer — React Dashboard

The frontend is a React single-page application featuring two primary views: a shipment list dashboard and an interactive map visualization powered by the Google Maps API. Both users and administrators access real-time shipment status, predicted delivery times, and rerouting notifications. The SPA is deployed to Amazon S3 static hosting behind a CloudFront distribution, delivering median page-load times under 700 milliseconds globally. All data is fetched via REST calls to the API Gateway endpoint, maintaining a clean separation between presentation and business logic.

### B. API Orchestration — Amazon API Gateway

Amazon API Gateway exposes five versioned REST endpoints: shipment submission, status query, delay alert retrieval, route update fetch, and analytics dashboard data. Each endpoint is backed by a dedicated Lambda function, enabling independent scaling. API Gateway handles request throttling at 300 requests per second per endpoint, JWT authorization via a Lambda authorizer, and CORS policy enforcement. All interactions are logged to Amazon CloudWatch Logs with a 30-day retention window.

### C. Processing Layer — Lambda and ML Delay Prediction

All shipment processing and ML inference runs inside Lambda functions. The ingestion Lambda is triggered by S3 PUT events via Amazon EventBridge when new shipment telemetry arrives. A dedicated ML prediction Lambda loads a pre-trained XGBoost model from S3 and scores each shipment record against features including route segment, historical delay rates, current traffic index, and weather severity. The model outputs a delay probability score; records

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

exceeding a 0.72 threshold trigger the self-correction Step Functions workflow. Provisioned concurrency of 3 instances on the prediction Lambda eliminates cold-start latency on this critical path.

**D. Orchestration Layer — Step Functions Self-Correction Loop**

The self-correction loop is LogiTrack's most architecturally significant contribution. Upon delay detection, the ML Lambda publishes an event to Amazon EventBridge, which triggers a Step Functions Express Workflow. The state machine executes three sequential states: (1) Delay Confirmation — a Lambda re-validates the delay signal against the latest telemetry to suppress false positives; (2) GenAI Route Planning — a Lambda invokes the GenAI logistics planner with the current shipment context, traffic data, and weather feed, receiving an optimized alternative route; (3) Data Update — a Lambda writes the new route to the Shipment Data table in S3 and pushes a real-time notification to the driver's dashboard. The entire workflow completes in a median of 4.1 seconds from delay detection to driver notification.

**E. Data Layer — S3, AWS Glue, and Apache Spark**

Amazon S3 serves as the primary data lake with three prefixes: raw/ for incoming shipment telemetry, processed/ for ML-scored records, and updated/ for rerouted shipment data. AWS Glue runs ETL jobs powered by Apache Spark that crawl the processed/ prefix, infer schemas, and populate an AWS Glue Data Catalog enabling ad-hoc SQL analytics through Amazon Athena. Spark's distributed processing engine handles 12,000 shipment records per minute on a G.1X worker cluster (4 vCPU, 16 GB RAM), with Glue job bookmarks ensuring incremental processing and preventing duplicate scoring.

## IV. GENAI ROUTE PLANNING AND SELF-CORRECTION LOGIC

**A. GenAI Logistics Planner**

When the Step Functions workflow reaches the GenAI Route Planning state, a Lambda function constructs a structured prompt containing: the current shipment's origin, destination, and current location; the predicted delay reason (traffic congestion, weather event, or road closure); real-time traffic data from the HERE Maps API; current weather conditions from OpenWeatherMap; and delivery time constraints. The GenAI planner — built on a large language model accessed via API — synthesizes a natural-language route narrative alongside a structured JSON waypoint sequence. The JSON output is parsed and stored directly in the S3 Updated Data prefix for immediate driver consumption.

**B. Self-Correction Logic Implementation**

The self-correction loop is designed for both reliability and speed. False positive suppression is handled in the Delay Confirmation state, which requires the delay probability to exceed 0.72 on two consecutive telemetry readings within a 90-second window before triggering rerouting. This dual-threshold approach reduces spurious rerouting events by 78% in testing. Once confirmed, the GenAI planner is invoked with a temperature of 0.3 to ensure deterministic, safe route recommendations rather than creative but impractical alternatives. The final state updates the Shipment Data table atomically, ensuring drivers always see a consistent view of their active route.

**C. Traffic and Weather Integration**

LogiTrack integrates two external data sources into the GenAI planning context. The HERE Maps Traffic API provides real-time congestion indices for each road segment along the planned route, updated every 2 minutes. The OpenWeatherMap API provides current weather severity scores for each geographic zone. Both feeds are cached in an ElastiCache Redis cluster with a 90-second TTL, reducing external API calls by 82% during high-frequency rerouting events while maintaining data freshness within acceptable bounds for logistics decision-making.

## V. IMPLEMENTATION DETAILS

The frontend React application is built with Vite 5.0, producing optimized static assets with average bundle sizes of 193 KB gzipped. The map visualization component uses the Google Maps JavaScript API with custom shipment status overlays rendered as GeoJSON layers, enabling administrators to monitor all active shipments simultaneously on a single map view. Shipment telemetry is pushed to the frontend via Amazon API Gateway WebSocket connections, enabling sub-second dashboard refresh without polling.

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

The ML delay prediction model is an XGBoost classifier trained on 2.4 million historical shipment records spanning 18 months, with features engineered from GPS telemetry, route metadata, weather history, and traffic patterns. The model is serialized with joblib and stored in S3, loaded into Lambda memory at cold start and cached for subsequent invocations. Feature engineering is performed within the Lambda function using pandas and scikit-learn pipelines, adding 0.3 seconds to prediction latency. Model retraining is scheduled weekly via AWS Glue, with the new model artifact automatically deployed to S3 and invalidated in Lambda's environment variable cache.

Security controls span the full stack. S3 bucket policies enforce server-side encryption with AWS KMS customer-managed keys and block all public access. Lambda execution roles follow least-privilege principles with separate IAM roles per function. Step Functions execution roles are scoped to invoke only the specific Lambda ARNs in each state. All inter-service communication occurs within a private VPC with interface endpoints for S3, DynamoDB, and API Gateway. AWS WAF protects the API Gateway endpoint with managed rule groups blocking OWASP Top 10 attack vectors. Driver data updates in the S3 Updated Data prefix use object-level CloudTrail logging for a full audit trail of all rerouting events.

The CI/CD pipeline is implemented with GitHub Actions. Commits to the main branch trigger a workflow that runs pytest unit tests, packages Lambda deployment archives, updates function code via AWS CLI, and deploys Step Functions state machine definitions via AWS CDK defined entirely in TypeScript. The full pipeline completes in under 6 minutes, enabling rapid iteration on both the ML model and GenAI prompt templates.

## VI. EXPERIMENTAL RESULTS

Evaluation used 800 shipment scenarios across urban, suburban, and rural route types, with ground-truth delay outcomes annotated from actual delivery records. A panel of 150 logistics professionals — 60 fleet managers, 50 dispatch coordinators, and 40 route planners — evaluated the system's rerouting recommendations across 200 delay-triggered scenarios.

### A. Delay Prediction and Route Planning Performance

**Table 2: System Performance Comparison**

| Metric | LogiTrack (Ours) | Baseline ML Only | Gain |
|---|---|---|---|
| Delay Prediction Accuracy | 93.6% | 76.4% | +17.2% |
| Route Reroute Success Rate | 89.1% | N/A | — |
| Avg. Delivery Time Reduction | 21.4% | 8.2% | +13.2% |
| Self-Correction Response Time | 4.1s | — | — |
| Data Pipeline Throughput | 12,000 rec/min | 4,200 rec/min | +185% |

LogiTrack's XGBoost delay prediction model achieves 93.6% accuracy on held-out test shipments, a 17.2 percentage point improvement over the baseline ML-only system that lacked real-time traffic and weather features ($t(799)=19.3$, $p<0.001$). The GenAI route planner's suggestions were accepted by logistics professionals in 89.1% of evaluated cases, with rejected routes most commonly cited as impractical for heavy vehicles due to bridge weight limits not captured in the traffic API.

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

**B. System Performance**

**Table 3: System Latency Metrics**

| Operation | Latency (p50) | Latency (p95) |
|---|---|---|
| Shipment Event Ingestion | 0.8s | 1.3s |
| ML Delay Prediction (Lambda) | 1.2s | 2.0s |
| GenAI Route Planning | 3.4s | 5.1s |
| End-to-End Self-Correction | 4.1s | 6.8s |
| Dashboard Data Refresh | 0.6s | 1.1s |
| System Uptime (30-day) | 99.8% | — |

**C. Feature Evaluation**

**Table 4: Feature-Level Performance**

| Feature | Score / Metric | User Rating (5pt) |
|---|---|---|
| ML Delay Prediction | Accuracy: 93.6% | 4.5 ± 0.6 |
| GenAI Route Planner | Acceptance Rate: 89.1% | 4.6 ± 0.5 |
| Self-Correction Automation | Trigger Accuracy: 91.3% | 4.4 ± 0.7 |
| Real-Time Dashboard | Refresh Latency: 0.6s | 4.3 ± 0.8 |
| Overall Platform | — | 4.5 ± 0.6 |

**D. Comparative Platform Analysis**

**Table 5: Comparative Platform Analysis**

| Feature | LogiTrack (Ours) | Oracle TMS | SAP Logistics |
|---|---|---|---|
| Serverless Architecture | Yes | No | No |
| ML Delay Prediction | Yes (93.6%) | Partial | Partial |
| GenAI Route Optimization | Yes | No | No |
| Self-Correction Logic | Yes | No | No |
| Query Latency (p95) | 6.8s | 12-18s | 15-22s |
| User Rating | 4.5/5 | 3.8/5 | 3.6/5 |

LogiTrack demonstrates 56–69% lower end-to-end self-correction latency than comparable commercial TMS platforms (Table 5), attributed to Lambda's provisioned concurrency and Step Functions' low-overhead state transitions. The self-correction capability — automated delay detection to driver route update without human intervention — remains absent from all evaluated commercial alternatives.

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

## VII. ERROR ANALYSIS AND SYSTEM ROBUSTNESS

LogiTrack's overall delay prediction error rate of 6.4% decomposes into feature staleness errors (2.8%), model boundary cases (2.1%), and false positives suppressed by the dual-threshold filter (1.5%). Feature staleness errors occurred when traffic API data lagged real conditions by more than 3 minutes during high-congestion periods — a known limitation of polling-based traffic feeds. Increasing the Redis cache TTL from 90 to 60 seconds reduced this error category by 34% in post-deployment testing.

GenAI route planning errors manifested as 10.9% of recommendations being rejected by logistics professionals. Analysis of rejected routes revealed three primary failure modes: bridge weight limit violations (42% of rejections), routes through restricted commercial vehicle zones (31%), and excessively long detours that exceeded the delivery time window (27%). Enriching the GenAI prompt with vehicle class constraints and restricted zone geofences is planned for the next system iteration.

Performance under load was evaluated with Apache JMeter simulating 300 concurrent shipment events. Median end-to-end self-correction latency degraded gracefully from 4.1 to 5.6 seconds, and the p99 remained below 11 seconds throughout the test. Lambda's automatic scaling provisioned 47 concurrent instances at peak load within 8 seconds, demonstrating the serverless architecture's elasticity for burst logistics scenarios such as peak holiday shipping periods.

## VIII. DISCUSSION

The results confirm that a fully serverless architecture is operationally superior to server-based alternatives for real-time logistics workloads. The Step Functions self-correction loop proved particularly impactful: automating the delay-to-reroute cycle eliminated the 8–15 minute human response window observed in baseline operations, compressing it to a median of 4.1 seconds. This latency reduction directly translates to delivery time improvements, as early rerouting avoids compounding delays that accumulate when a congested segment is traversed rather than bypassed.

The GenAI route planner's 89.1% professional acceptance rate validates the approach of providing rich contextual inputs — traffic, weather, vehicle constraints, and delivery windows — rather than routing based on distance alone. The 10.9% rejection rate highlights the importance of domain-specific constraint encoding: geospatial knowledge of vehicle restrictions that is implicit in an experienced dispatcher's mental model must be explicitly represented in the prompt or retrieved from a constraints database.

Limitations include: (1) the traffic API polling interval creates a 90-second data freshness lag; (2) the GenAI planner lacks vehicle class geofencing awareness; (3) the XGBoost model requires weekly retraining to capture seasonal route patterns; (4) GenAI API costs of approximately $0.003 per rerouting event scale linearly with delay frequency. Future work will explore replacing polling with event-driven traffic webhooks, integrating HERE Map Attributes for vehicle restriction data, and evaluating fine-tuned logistics-domain LLMs to reduce per-call API costs.

## IX. CONCLUSION AND FUTURE WORK

This paper presented LogiTrack, a cloud-native supply chain logistics optimization platform that unifies ML-powered delay prediction, GenAI route planning, and automated self-correction within a fully serverless AWS architecture. The system achieves 93.6% delay prediction accuracy, 89.1% route-reroute acceptance, and a 21.4% reduction in average delivery time across 800 evaluated shipment scenarios. The self-correction loop — implemented via Step Functions — reduces the delay-to-driver-notification latency to a median of 4.1 seconds, eliminating the human intervention bottleneck in existing logistics workflows.

The Apache Spark-based Glue ETL pipeline processes 12,000 shipment records per minute, enabling the analytics layer to provide fleet managers with near-real-time operational visibility. The React dashboard with integrated map visualization received a user rating of 4.5/5.0, confirming strong end-user adoption of the real-time monitoring interface.

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Future work will: (1) integrate HERE Map Attributes for vehicle class geofencing in the GenAI planner prompt; (2) replace traffic API polling with event-driven webhooks to eliminate the 90-second freshness lag; (3) evaluate fine-tuned logistics-domain LLMs for reduced per-rerouting API costs; (4) implement multi-modal delivery optimization supporting mixed fleets of trucks, vans, and drones; (5) develop a federated ML training pipeline that learns from rerouting outcomes across multiple logistics operators while preserving data privacy.

## X. ACKNOWLEDGMENT

The authors gratefully acknowledge the guidance of Mr. Arun Prasad from the Department of Artificial Intelligence and Data Science, Sri Manakula Vinayagar Engineering College, Puducherry. Sincere thanks to the 150 logistics professionals who participated in the evaluation study and to the domain experts who annotated the delay ground-truth dataset used for model validation.

## REFERENCES

[1] A. Gunasekaran, C. Patel, and R. E. McGaughey, "A framework for supply chain performance measurement," Int. J. Production Economics, vol. 87, no. 3, pp. 333–347, 2004.
[2] L. Chen, Z. Wang, and Y. Liu, "Machine learning-based shipment delay prediction for last-mile logistics," IEEE Trans. Intell. Transp. Syst., vol. 22, no. 8, pp. 5041–5052, 2021.
[3] J. Park and S. Kim, "Serverless real-time shipment tracking with AWS Lambda," in Proc. IEEE Cloud Comput. Conf., 2022, pp. 78–85.
[4] R. Sharma, P. Gupta, and M. Singh, "Generative AI for dynamic route optimization in urban logistics," in Proc. ACM SIGKDD, 2023, pp. 412–421.
[5] T. Nguyen, H. Tran, and B. Le, "Scalable ETL for logistics data lakes using AWS Glue and Apache Spark," J. Big Data, vol. 9, no. 1, p. 47, 2022.
[6] Amazon Web Services, "Orchestrating microservices with AWS Step Functions," AWS Whitepaper, 2023. [Online]. Available: https://aws.amazon.com/step-functions/
[7] D. Ivanov, A. Dolgui, and B. Sokolov, "The impact of digital technology and Industry 4.0 on the ripple effect and supply chain risk analytics," Int. J. Production Research, vol. 57, no. 3, pp. 829–846, 2019.
[8] M. Christopher, Logistics and Supply Chain Management, 5th ed. Harlow, UK: Pearson Education, 2016.
[9] J. Brown, T. Smith, and K. Patel, "Real-time route optimization with large language models," arXiv:2401.05123, 2024.
[10] AWS, "Amazon EventBridge: Event-driven architecture at scale," AWS Documentation, 2024. [Online]. Available: https://docs.aws.amazon.com/eventbridge/
[11] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in Proc. ACM SIGKDD, 2016, pp. 785–794.
[12] R. Ballou, Business Logistics/Supply Chain Management, 5th ed. Upper Saddle River, NJ, USA: Prentice Hall, 2004.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

# INTERNATIONAL JOURNAL OF MULTIDISCIPLINARY RESEARCH

## IN SCIENCE, ENGINEERING AND TECHNOLOGY

| Mobile No: +91-6381907438 | Whatsapp: +91-6381907438 | ijmrset@gmail.com |

www.ijmrset.com