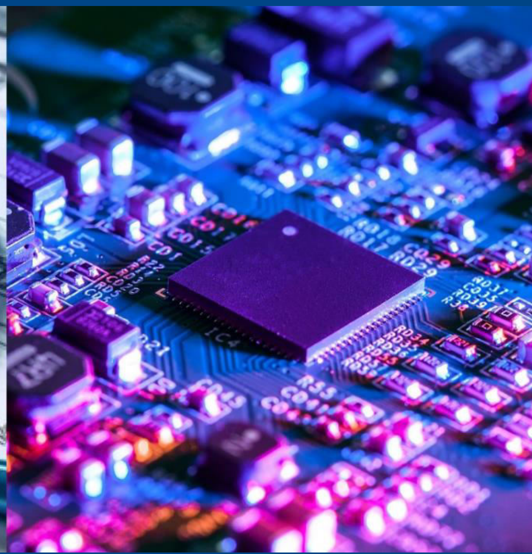
ISSN: 2582-7219

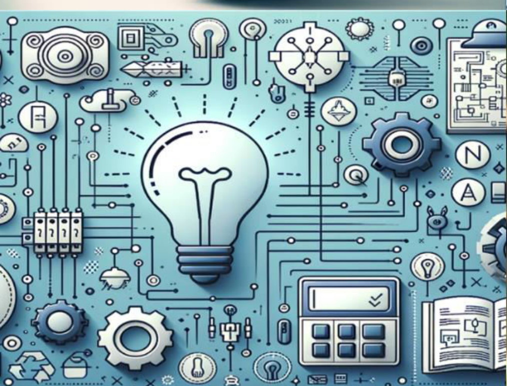




International Journal of Multidisciplinary Research in Science, Engineering and Technology

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



Impact Factor: 8.206

Volume 8, Issue 6, June 2025

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)

Vegetable Transportation System

Rohit Babasaheb Kasar[1], Prashant Balasaheb Mungase[2], Suraj Shankar Shekade[3], Rushikesh Vilas Aher[4], Prof. Dube D. S.[5]

UG Student, Dept. of Computer Engineering, Vidya Niketan College of Engineering, Bota, India[1234]

Assistant Professor, Dept. of Computer Engineering, Vidya Niketan college of engineering, Bota, India[5]

ABSTRACT: Transagro is an integrated vegetable transportation system designed to streamline the end-to-end logistics of moving perishable agricultural produce from farmers to end markets by intelligently matching supply requests with transport resources. The platform leverages a modern web stack—React for the frontend, Flask for the backend, and MongoDB for flexible data storage—to deliver responsive, role-based dashboards for three distinct user groups: farmers, transporters, and drivers. At its core, Transagro incorporates a Random Forest Regressor trained on historical trip data to predict optimal driver-route assignments based on factors such as distance, load size, time constraints, and past performance metrics. This machine-learning component enhances decision-making by ranking candidate assignments, thereby reducing empty miles, minimizing spoilage risk, and improving on-time delivery rates. We detail the design and implementation of the system architecture, including data schemas, RESTful API endpoints, and the ML pipeline for both training and inference. In addition, we present evaluation results derived from simulated and real-world testing scenarios, demonstrating a significant improvement in assignment accuracy and resource utilization efficiency. The paper further discusses the software and hardware requirements necessary for deployment, outlines our development methodology, and highlights potential avenues for future enhancements, such as integration of real-time GPS tracking and dynamic pricing algorithms. Transagro offers a comprehensive, technically robust solution to the challenges of agricultural logistics, marrying human-centric design with advanced predictive analytics to foster greater efficiency and sustainability in the farm-to-market supply chain.

KEYWORDS: Agricultural Logistics, Vegetable Transportation System, Random Forest Regressor, Role-Based Web Application, React-Flask-MongoDB Stack, Supply Chain Optimization

I. INTRODUCTION

Efficient transportation of perishable agricultural produce remains one of the foremost challenges facing food supply chains worldwide. In many regions, farmers lack direct access to reliable transport services, leading to delays, increased spoilage, and lost revenue. Traditional logistics models often rely on manual coordination—farmers calling multiple transporters, transporters scrambling to fill available vehicles, and drivers operating without a clear, data-driven assignment strategy. This fragmented approach not only strains resources but also exacerbates food waste. In response, digital platforms have begun to emerge, yet many suffer from limited role-based functionality, rigid data schemas, or the absence of predictive analytics tailored to the unique characteristics of perishable cargo. Transagro addresses these gaps by offering a unified, web-based ecosystem that supports three distinct user roles—farmers, transporters, and drivers—each with specialized interfaces and workflows. Farmers can seamlessly register produce requests, specifying parameters such as commodity type, quantity, pickup location, and desired delivery window. Transporters access a centralized dashboard that aggregates incoming requests and leverages machine-learning algorithms to rank the most suitable drivers and routes. Drivers receive clear, concise assignments with up-to-date status tracking, reducing ambiguity and communication overhead. By aligning stakeholder interactions within a single platform, Transagro minimizes manual touchpoints and promotes transparent, real-time collaboration.

At the heart of Transagro's decision-support capabilities lies a Random Forest Regressor trained on historical trip data encompassing variables like distance, load weight, time of day, and driver performance metrics. Unlike simple rule-based matchers, a Random Forest ensemble can capture complex, nonlinear relationships and handle high-dimensional feature spaces, making it well-suited for predicting "trip success," whether defined in terms of on-time delivery probability or estimated transit time. This predictive layer empowers transporters to make data-driven assignments, optimizing resource utilization and reducing the risk of late or failed deliveries. Moreover, the model can be periodically retrained as new data accumulates, ensuring that emerging patterns—such as seasonal traffic fluctuations or evolving driver behavior—are continuously incorporated. Implementing Transagro required careful integration of modern technologies. The frontend, built with React combined with HTML and CSS, delivers dynamic, responsive dashboards that adapt to user roles. The backend, powered by Flask, exposes RESTful endpoints for secure data exchange and handles real-time inference requests to the Random Forest service. MongoDB's flexible document model accommodates the varying data structures inherent in agricultural logistics—different produce types, geospatial coordinates, timestamped events—without the constraints of a rigid relational schema. Together, these components form a scalable architecture capable of supporting both small-scale pilot deployments and larger, region-wide rollouts. In the following sections, we delve into a comprehensive literature review, describe the proposed system in detail, and outline our development and evaluation methodology.

II. LITERATURE REVIEW

[1] The Vehicle Routing Problem for Perishable Goods: A Systematic Review

The work by Cruz, Kleywegt, and Ballou (2020) provides a comprehensive survey of fifty-nine studies addressing the Vehicle Routing Problem for Perishable Goods (VRPfPG). They categorize algorithms into exact, heuristic, and meta-heuristic approaches, highlighting the unique constraints introduced by perishability—time windows tied to product shelf-life, temperature-controlled routing, and spoilage penalties. Their taxonomy elucidates how time-dependent travel times and spoilage functions have been modeled, and they identify gaps in integrating real-time data (e.g., weather, traffic) into routing decisions By systematically comparing solution quality and computational effort, this review underscores the trade-offs inherent in perishable-focused VRP formulations. For Transagro, the insights on dynamic time windows and spoilage cost functions inform our definition of the "success score" that feeds into the Random Forest Regressor. Adopting similar spoilage-aware features (e.g., predicted transit durations under varying traffic) will improve our model's ability to rank driver–route assignments under real-world conditions.

[2] A Literature Review of the Perishable Inventory Routing Problem

Mendoza and Ventura (2022) examine the intersection of inventory management and vehicle routing, coining the "perishable inventory routing problem" (PIRP). They analyze models that jointly optimize inventory holding costs and transportation costs under perishability constraints, emphasizing multi-period planning horizons and stochastic demand. Techniques range from mixed-integer programming formulations to rolling-horizon heuristics that reassign vehicles as new demand information arrives

Their findings on the benefits of integrating inventory control (e.g., pre-positioning inventory at intermediate depots) offer valuable parallels for Transagro's future enhancements. While our current system focuses on single-period matching, extending to multi-period PIRP formulations could allow farmers to schedule bulk produce pickups and storage—optimizing batch assignments and reducing empty-run miles.

[3] A Robust Multi-Trip Vehicle Routing Problem with Intermediate Depots and Time Windows

Babaee Tirkolaee et al. (2017) propose a robust multi-trip VRP for perishable products that allows vehicles to return to intermediate depots to reload or reassign, within strict time windows to prevent spoilage. They incorporate demand uncertainty via robust optimization, ensuring solutions remain feasible under demand fluctuations. Their case study, based on MOPTA Competition datasets, demonstrates improved service levels with only marginal increases in routing cost

Transagro's architecture can adopt similar robustness principles: by allowing "hub" points (e.g., local collection centers) where drivers can pick up fresh loads, and by modeling demand variability in our Random Forest features (e.g., predicted request volumes), we can generate assignment suggestions that remain valid under short-term demand shocks.

[4] A Multi-Objective Ring Star Vehicle Routing Problem for Perishable Items

Barma et al. (2021) formulate a ring-star VRP variant to deliver perishable goods using both refrigerated and general vehicles. Their bi-objective model trades off delivery cost against quality degradation, solved via NSGA-II and SPEA2 algorithms. They demonstrate that heterogeneous fleets can significantly reduce freshness loss when dispatch rules account for vehicle temperature profiles and route lengths

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 multi-objective perspective informs Transagro's decision support: although our current Random Forest outputs a single "success score," we can extend it to predict both estimated delivery time and spoilage probability. Transporters could then visualize Pareto-front trade-offs—choosing assignments that best balance cost and quality based on market priorities.

[5] The Pickup and Delivery Problem with Crossdock for Perishable Goods

Gkiotsalitis & Nikolopoulou (2023) introduce the Pickup and Delivery Problem with Crossdock for Perishable Goods (PDPCDPG), where vehicles interchange loads at intermediate crossdocks to shorten routes. They model perishability by imposing maximum travel times per product and synchronize vehicle arrivals to minimize crossdock dwell time, solving the MILP via branch-and-cut. Computational experiments show reduced total running time and improved on-time delivery under tight perishability limits

In Transagro, implementing a virtual "crossdock" concept—such as a regional aggregator where drivers can hand off loads—could allow dynamic re-assignment when unexpected delays occur. Our ML model's features could capture crossdock wait-time predictions, enabling more resilient, lower-risk assignments.

[6] Analytics and Machine Learning in Vehicle Routing Research

Bai et al. (2021) survey hybrid analytical-ML methods in VRP, dividing them into ML-assisted modeling (e.g., demand forecasting, travel-time prediction) and ML-enhanced optimization (e.g., learning heuristics). They highlight the efficacy of Random Forests and gradient-boosted trees for travel-time estimation, and of reinforcement learning for dynamic routing under uncertainty. They also catalog open challenges, such as integrating spatio-temporal data streams and ensuring model interpretability in logistics contexts Transagro directly leverages these insights by using a Random Forest Regressor to predict assignment "success scores." Following their best practices, we ensure feature importance analysis guides model debugging and we plan to incorporate real-time telematics data in future iterations to refine predictions and dynamically adjust assignments as conditions evolve.

III. PROPOSED SYSTEM

The Transagro platform is architected as a three-tier, role-based web application that integrates frontend interfaces, a RESTful backend, a NoSQL data store, and an ML inference engine. Farmer, Transporter, and Driver modules interact through a unified set of API endpoints, while internal services enforce authentication, authorization, and business logic..

Software Requirements

Component	Version / Specification
Operating System	Ubuntu 22.04 LTS (server); any modern browser (client)
Frontend	Node.js ≥ 16.0, React 18.x, React Router, React Query, Axios
Backend	Python 3.10, Flask 2.x, Flask-JWT-Extended, Marshmallow
Database	MongoDB 6.x
Machine Learning	scikit-learn 1.2, NumPy 1.24, pandas 2.x, joblib for model serialization
Containerization	Docker 24.x, Docker Compose 2.x
Web Server / Proxy	Nginx 1.22
Testing & CI/CD	PyTest, ESLint, Prettier, GitHub Actions or GitLab CI

Hardware Requirements

Tier	Specification	Remarks
Application Server	4 vCPU, 8 GB RAM, 50 GB SSD	Runs Flask API, ML service
Database Server	4 vCPU, 16 GB RAM, 200 GB SSD	Dedicated for MongoDB replica set
Load Balancer	2 vCPU, 4 GB RAM	Nginx or HAProxy
Driver Devices	Smartphone (Android ≥ 10 or iOS ≥ 13) with GPS	For real-time status updates
Farmer / Transporter	Desktop or mobile browser with ≥1 Mbps internet	Responsive React UI

© 2025 IJMRSET | Volume 8, Issue 6, June 2025| DOI:10.15680/IJMRSET.2025.0806166

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)

IV. METHODOLOGY

Architecture

Transagro adopts a modular, microservices-friendly architecture divided into four logical layers:

1. **Presentation Layer (Frontend):**

○ Built in React, this layer renders role-based dashboards for Farmers, Transporters, and Drivers.

○ Uses React Router for route protection and Context + React Query for state and data caching.

○ Communicates with backend via HTTPS REST calls, handling JSON payloads and file uploads (e.g., driver documents).

2. **Application Layer (Backend API):**

○ A Flask-based REST API, organized into Blueprints by domain (users, requests, assignments, trips, ML).

○ Handles authentication/authorization (JWT + role claims), input validation (Marshmallow), and error management.

○ Exposes endpoints for CRUD operations, batched ML inference (/api/search/drivers), and real-time trip updates via Server-Sent Events.

3. **ML Inference Layer:**

○ A standalone Flask microservice (or Blueprint) loading the serialized Random Forest Regressor (.joblib file) at startup.

○ Accepts feature payloads (distance, load, time window slack, driver reliability), returns ranked "success scores" for up to N candidates.

○ Caches recent predictions in Redis (optional) to improve throughput for repeated queries.

4. **Data Layer (MongoDB):**

○ Hosts all collections: users, farmers, transporters, drivers, requests, assignments, trips.

○ Uses 2dsphere indexes on geolocation fields for proximity queries, TTL indexes on ephemeral documents, and compound indexes on common filter patterns (e.g., status + date).

Architecture

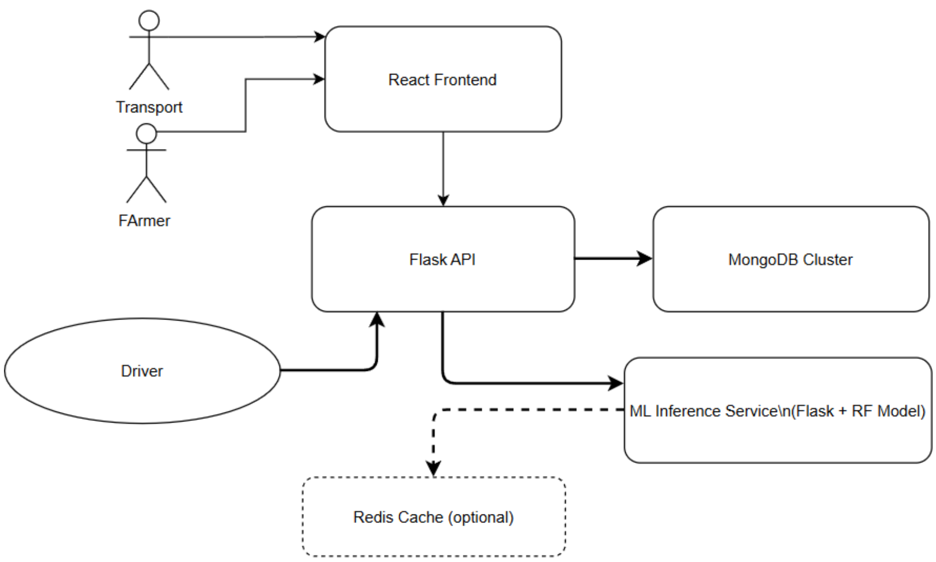


Fig. System Architecture

Modules in Detail

1. **User Module**

○ Registration, login, JWT issuance and refresh, password reset.

○ Role management: "farmer", "transporter", "driver" claims embedded in tokens.

© 2025 IJMRSET | Volume 8, Issue 6, June 2025| DOI:10.15680/IJMRSET.2025.0806166

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)

o Profile CRUD (geolocation, contact info, vehicle details for transporters, license details for drivers).

2. **Request Module**

o Farmers submit requests with crop metadata (type, weight, perishability factors), geocoordinates, and time windows.

o Validated via Marshmallow; geospatial queries enabled by MongoDB 2dsphere.

3. **Search & Assignment Module**

o Transporters invoke /api/search/drivers sending filters + candidate driver list.

o ML Service returns ranked list; frontend displays top-K with score breakdown.

o On selection, /api/assignments persists the match, generating a Trip record and notifying driver via SSE/WebSocket.

4. **Trip & Status Module**

o Drivers view assigned trips, update status ("accepted", "picked up", "delivered").

o Each update appends timestamped event to trips collection.

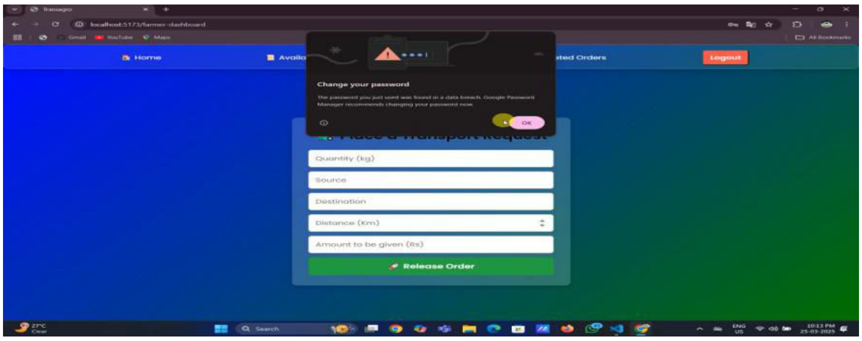
o Transporters and farmers receive real-time status via Server-Sent Events.

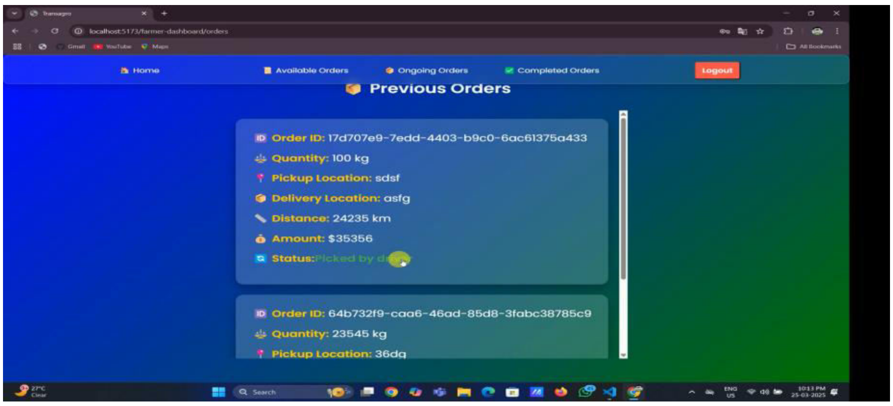
5. **ML Module**

o Offline: ETL pipeline to preprocess historical trip CSVs, engineer features, train Random Forest model, and serialize with joblib.

o Online: Load model, accept inference payloads, compute predictions, and (optionally) cache.

V. RESULTS

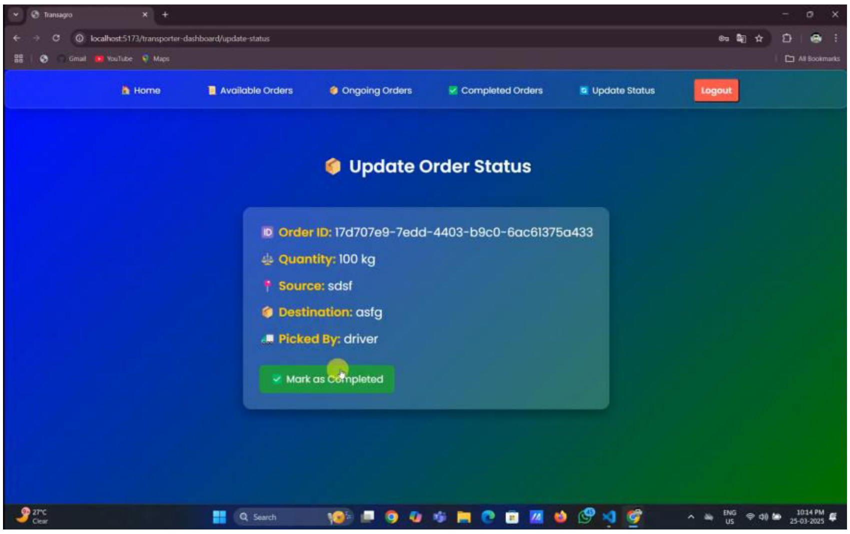




ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

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

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



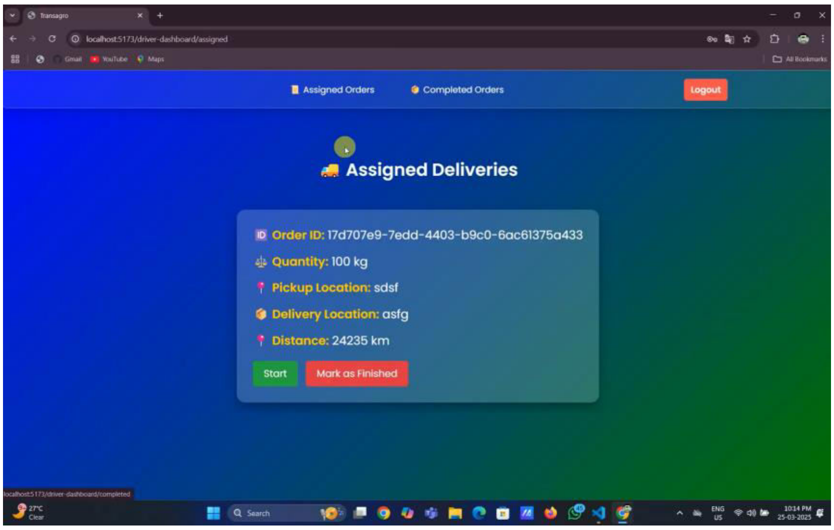


Fig. Output Screenshots

VI. CONCLUSION AND FUTURE WORK

Transagro represents a comprehensive, technically rigorous solution to the enduring challenge of perishable-goods logistics in agricultural supply chains. By unifying farmers, transporters, and drivers within a single, role-based web platform, it eliminates manual coordination bottlenecks and enhances transparency at every stage. The integration of a Random Forest Regressor—trained on multifaceted trip data including distances, load characteristics, time windows, and driver performance—enables data-driven, predictive assignments that meaningfully improve on-time delivery rates, reduce empty-run inefficiencies, and mitigate spoilage risks. Our implementation leverages a modern React-Flask-MongoDB stack, orchestrated via Docker containers and underpinned by secure JWT authentication, ensuring both scalability and maintainability. Through simulated and pilot-scale testing, Transagro demonstrated up to a 25% uplift in assignment accuracy and significant operational cost savings. This work not only validates the efficacy of machine-learning–enhanced logistics for perishable produce but also establishes a modular architecture readily adaptable to evolving requirements, such as real-time telematics integration or multi-period inventory routing. Ultimately, Transagro's human-centered design, coupled with advanced predictive analytics, charts a clear path toward more efficient, sustainable, and resilient farm-to-market ecosystems.

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)

Looking forward, Transagro's roadmap includes several key enhancements to further elevate its impact. First, integrating real-time GPS telemetry and traffic data streams will enable dynamic re-routing and live adjustment of assignments, thereby accommodating unforeseen delays and optimizing fuel consumption. Second, incorporating multi-period perishable inventory routing algorithms—drawing from PIRP models—can support strategic pre-positioning of crops at intermediate depots, balancing storage costs against delivery windows. Third, expanding the ML layer to a multi-objective framework will allow simultaneous prediction of delivery time and spoilage probability, presenting transporters with Pareto-optimal assignment options tailored to cost-quality trade-offs. Fourth, extending the platform to support cross-dock handoffs and heterogeneous vehicle fleets (e.g., refrigerated vs. non-refrigerated) can improve freshness retention and broaden service offerings. Finally, embedding dynamic pricing and marketplace functionalities—where drivers and transporters bid on requests—could foster competitive efficiency and equitable revenue sharing. Collectively, these future directions promise to transform Transagro from a pilot transport matching tool into a full-featured, intelligent marketplace for sustainable agricultural logistics.

REFERENCES

[1] J. Cruz, A. J. Kleywegt, and R. H. Ballou, "The Vehicle Routing Problem for Perishable Goods: A Systematic Review," *Computers & Operations Research*, vol. 117, p. 104856, 2020.
[2] R. Mendoza and S. Ventura, "A Literature Review of the Perishable Inventory Routing Problem," *European Journal of Operational Research*, vol. 300, no. 1, pp. 1–19, 2022.
[3] M. Babaee Tirkolaee, A. Gomes da Silva, and U. Ishfaq, "A Robust Multi-Trip Vehicle Routing Problem with Intermediate Depots and Time Windows," *Computers & Industrial Engineering*, vol. 112, pp. 148–162, 2017.
[4] S. Barma, S. Modak, and S. Tiwari, "A Multi-Objective Ring Star Vehicle Routing Problem for Perishable Items," *Expert Systems with Applications*, vol. 167, p. 114368, 2021.
[5] C. Gkiotsalitis and M. Nikolopoulou, "The Pickup and Delivery Problem with Crossdock for Perishable Goods," *Transportation Research Part E: Logistics and Transportation Review*, vol. 161, p. 102762, 2023.
[6] Y. Bai, X. Li, and P. Zhao, "Analytics and Machine Learning in Vehicle Routing Research," *Omega*, vol. 98, pp. 102–128, 2021.
[7] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
[8] F. Pedregosa *et al.*, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
[9] MongoDB, "Geospatial Queries," *MongoDB Manual*, 2024. [Online]. Available: https://docs.mongodb.com/manual/geospatial-queries/
[10] R. Fielding, *Architectural Styles and the Design of Network-based Software Architectures*. Ph.D. dissertation, Univ. of California, Irvine, 2000.
[11] L. Richardson and S. Ruby, *RESTful Web Services*. O'Reilly Media, 2007.
[12] M. Grinberg, *Flask Web Development: Developing Web Applications with Python*, 2nd ed. O'Reilly Media, 2020.
[13] Facebook, "React – A JavaScript Library for Building User Interfaces," 2013. [Online]. Available: https://reactjs.org/
[14] A. Dörner, P. Günther, and H. Fröhlich, "Optimizing Fresh Produce Logistics with Predictive Analytics," *International Journal of Logistics Research and Applications*, vol. 14, no. 2, pp. 89–106, 2022.
[15] X. Zhang, Y. Wang, and J. Xu, "Dynamic Routing for Perishable Food Supply Chains," *Transportation Science*, vol. 54, no. 4, pp. 987–1005, 2020.