iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** X **Month of publication:** October 2026

DOI: https://doi.org/10.22214/ijraset.2026.85374

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue X Oct 2026- Available at www.ijraset.com

AI-Based College Transportation Management

S. Yoheswari[1], Mahadeer M[2], Kishore K R[3]
Department of Computer Science and Engineering, K.L.N College of Engineering (An Autonomous Institution, Affiliated to Anna University, Chennai), Tamil Nadu, India

Abstract: ***Colleges that run their own buses usually plan transport using fixed routes and estimated student counts. In practice, the number of students who actually travel changes from day to day, so some buses run with many empty seats while others are overcrowded. This paper presents the design of the AI-Based College Transportation Decision Support System, a full-stack web application that collects daily travel confirmations from students and uses rule-based software agents to analyse route-wise demand, check route compatibility, recommend bus allocation, and suggest seat optimization within capacity limits. Every recommendation is accompanied by a plain-language explanation, and no change becomes final until the administrator approves it. The previous allocation stays in force until a replacement plan is approved. The system is built with React.js, Node.js, Express.js and MongoDB, with JWT-based authentication and role-based access control. It uses predefined rules and calculations only; it does not use machine learning, GPS tracking, or external generative AI services. This paper describes the problem, the system architecture, the database design, the rule-based algorithms, and an illustrative worked example. Experimental evaluation on real college data is identified as future work.***

Keywords: College Transportation, Decision Support System, Rule-Based Agents, Bus Allocation, Seat Optimization, Route Compatibility, MERN Stack, Explainable Recommendations

I. INTRODUCTION

Many colleges provide bus service so that students and staff can reach campus from different parts of a city or district. Bus allocation is commonly based on fixed routes and an estimate of how many students use each route. However, the number of students who travel by bus is not constant. Absences, holidays, examinations, and changes in personal plans mean that the real demand on each route can differ from the estimate on any given day.

This mismatch causes two problems. One bus may carry only a few students and run with many empty seats, while another bus on a busy route may have more students than seats. Administrators usually discover this late and handle it manually, with little data to support their decisions.

This paper proposes the AI-Based College Transportation Decision Support System. Students confirm each evening whether they will use the college bus the next day. The system then analyses these confirmations, checks which routes can safely share a bus, and recommends allocations that make better use of available seats without exceeding capacity. The administrator reviews each recommendation and decides whether to approve, modify, or reject it.

The system is a decision-support tool. It recommends, explains, and waits for human approval. It is implemented with predefined rules and calculations in JavaScript and is therefore described as a rule-based AI decision-support system, not as a trained machine learning system.

A. Objectives of the Study

- To collect daily travel confirmations from students through a secure web portal.
- To calculate route-wise demand and identify routes with high and low demand.
- To recommend bus allocation based on demand, seating capacity, and bus availability.
- To check route compatibility using predefined rules before any redistribution is suggested.
- To suggest seat optimization without exceeding the capacity of any bus.
- To give an understandable explanation for every recommendation.
- To let the administrator approve, modify, or reject recommendations, and to notify students of final arrangements.

B. Scope

The system covers daily next-day planning for a single college with a defined set of buses, routes, and approved stops. It does not perform live vehicle tracking, route generation from maps, or demand prediction using machine learning.

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue X Oct 2026- Available at www.ijraset.com

II. REVIEW OF LITERATURE

Transportation planning has been studied extensively under the heading of vehicle routing. The truck dispatching problem was introduced by Dantzig and Ramser [2], and later work has produced a large body of methods, summarised in the book by Toth and Vigo [3]. These studies mainly concern how to build efficient routes for vehicles, which is a different question from the one addressed in this paper, namely how to adjust the use of an existing fixed route set from day to day.

The school bus routing problem has been reviewed by Park and Kim [4], who describe how stop selection, route construction, bus scheduling, and student assignment can be treated as related planning problems. Their review shows that capacity limits and acceptable travel conditions are central constraints in student transportation, which is consistent with the rules used in the present work.

Recent route optimization work, such as the study by Puneeth and Jayaprakash [1], examines how technologies like GPS tracking, real-time traffic data, and heuristic algorithms are adopted in logistics and urban transit. That study is based on a survey of transportation professionals and focuses on logistics organizations. The present work differs in scope. It addresses a college setting, uses student confirmations rather than GPS or traffic data, and relies on transparent rules rather than data-driven models.

III. RESEARCH GAP AND PROPOSED CONTRIBUTION

Much of the available literature concentrates on building new routes or on logistics operations. Less attention is given, in the sources reviewed by the authors, to small institutional bus services where the routes are fixed, but daily demand varies and an administrator must decide how to use seats efficiently. Such institutions often cannot afford GPS hardware or complex optimization software.

The contribution of this work is a practical, low-cost design that combines the following features in one web application. These are proposed project contributions, and the authors do not claim that they are novel in academic research, since no systematic literature search has been completed.

- Daily travel confirmation instead of reliance on fixed estimates.
- Rule-based software agents for structured transportation decisions.
- Route compatibility checking before redistribution.
- Seat optimization that never exceeds capacity.
- Explainable recommendations.
- Mandatory administrator approval before changes are finalized.
-

IV. SYSTEM DESIGN AND METHODOLOGY

A. System Architecture

The system has a student portal and an administrator portal, both served by a React.js frontend. The frontend communicates with a Node.js and Express.js backend through REST APIs. The backend first passes each request through an authentication module that verifies the JSON Web Token (JWT) [5] and the user role. It then calls transportation services, which run the rule-based agents and read or write data in MongoDB. The overall architecture is shown in Fig. 1.

The processing stages are: request from the student or administrator portal, REST API call, authentication and role check, transportation services, Demand Analysis and Bus Allocation Agents, Seat Optimization Agent, Recommendation Agent, administrator review and final approval, and student notification. These are logical stages inside one backend application. They are not separate servers or independent AI services.

©IJRASET: All Rights are Reserved

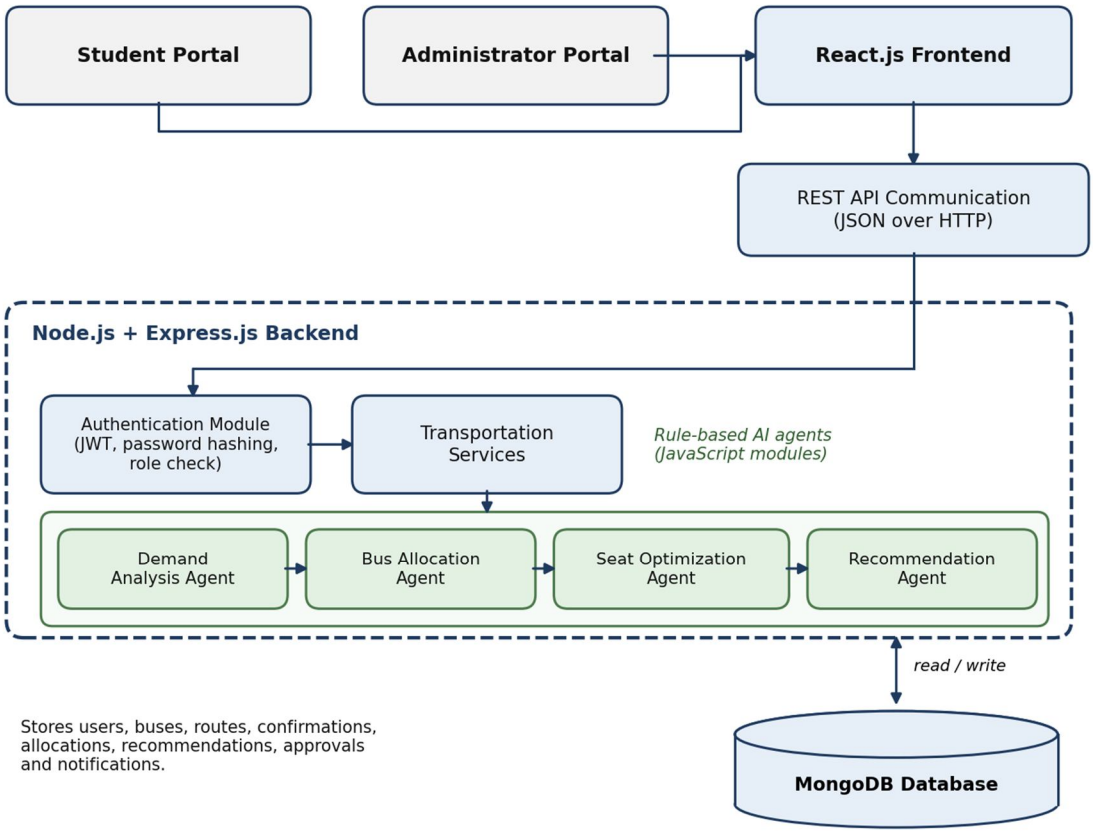


Fig. 1 System architecture of the proposed AI-Based College Transportation Decision Support System

B. User Roles

1) *Student:* A student registers and logs in, views routes, confirms travel for the next day, views the finalized bus allocation, and receives notifications.
2) *Administrator:* The administrator manages students, buses, routes and capacities, views confirmations and occupancy, runs the recommendation process, reviews explanations, approves, modifies or rejects recommendations, and notifies affected students. The daily workflow is shown in Fig. 2.

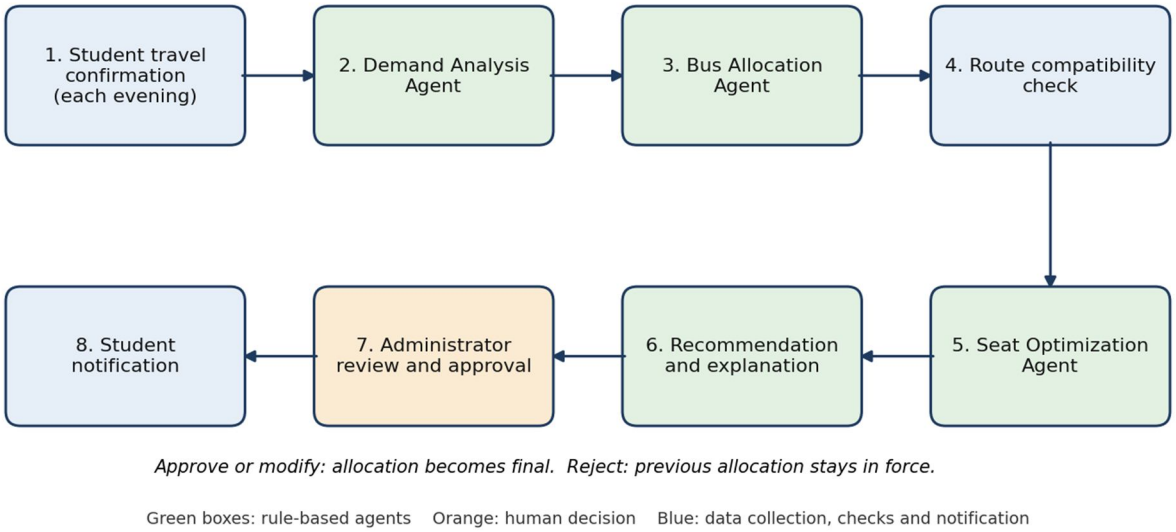


Fig. 2 Daily transportation decision workflow, from student confirmation to student notification

C. Database Design

The system uses MongoDB. The main collections are listed in Table I and their references are shown in Fig. 3.

TABLE I
MONGODB COLLECTIONS

Collection	Purpose	Key fields (proposed)
Users	Student and administrator accounts	name, email, passwordHash, role, routeId, stopId
Buses	Bus details and status	busNumber, capacity, status
Routes	Route names and approved stops	name, stops[], defaultBusId, compatibleRouteIds[]
TravelConfirmations	Travel decision for a date	studentId, travelDate, willTravel, submittedAt
Allocations	Student-to-bus assignment	studentId, busId, travelDate, status
Recommendations	Generated plans and explanations	travelDate, currentPlan, proposedPlan, explanation, status
Approvals	Administrator decisions	recommendationId, adminId, decision, remarks, decidedAt
Notifications	Messages to students	userId, message, readStatus, createdAt

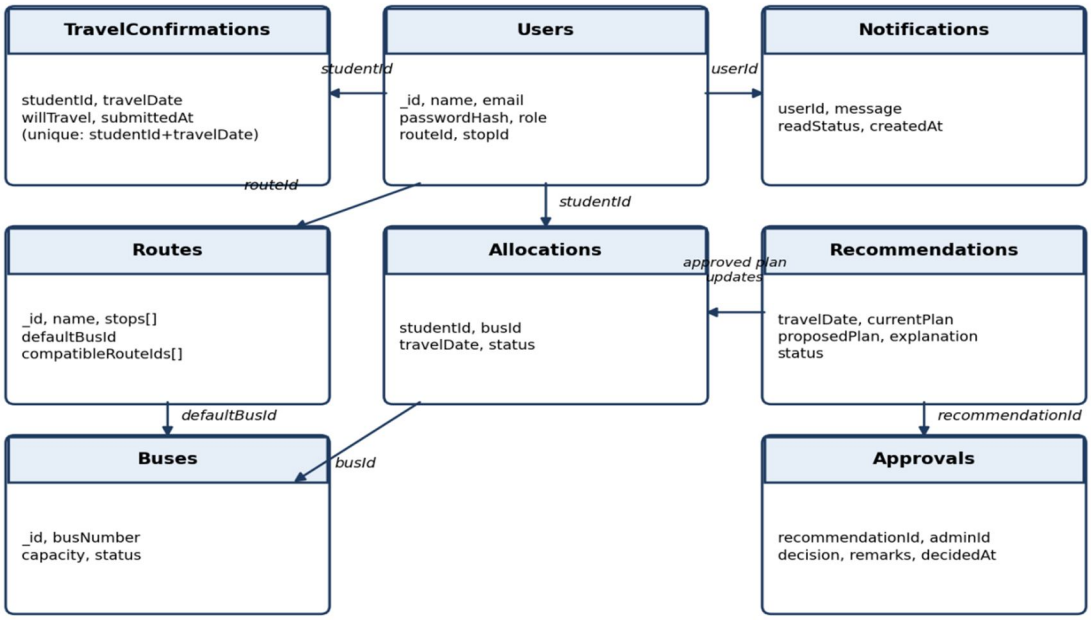


Fig. 3 MongoDB collections and the main references between them

A unique index on studentId and travelDate in TravelConfirmations prevents duplicate confirmations for the same day. Passwords are stored only as hashes. Routes are referenced by identifier from other collections so that route details are not duplicated.

D. Business Rules

The following rules are enforced by the backend:

1) A bus must never exceed its seating capacity.
2) Only valid, current confirmations are counted.
3) Route compatibility must be confirmed before students are moved between routes.
4) Every student must have a valid assignment.
5) Bus availability is considered in every plan.
6) Final allocations require administrator approval, and the previous allocation remains in force until then.
7) Every recommendation includes an explanation.
8) If no safe optimization exists, the system reports this and makes no change.
9) Changes must not compromise the next morning's transport arrangements.

E. Rule-Based Agents

1) *Demand Analysis Agent:* It counts valid confirmations for the target date and calculates, for each route, the confirmed demand, the capacity of the assigned bus, the available seats, and a demand status (over capacity, within capacity, or low).
2) *Bus Allocation Agent:* It compares each route's demand with the capacity of its assigned bus and with the list of available buses. If a spare bus is available, it can propose assigning it to an over-capacity route.
3) *Route Compatibility Check:* Compatibility is decided by predefined rules stored with the routes, such as shared stops, route overlap, approved pickup and drop-off points, and acceptable deviation set by the college. A pair of routes is compatible only if the administrator has defined it as so. Having empty seats is never sufficient.
4) *Seat Optimization Agent:* For each over-capacity route, it looks for a compatible route with free seats and proposes moving only as many eligible students as needed, never more than the free seats.
5) *Recommendation Agent:* It assembles the current plan, the proposed plan, students per bus, capacity, free seats, affected routes, the reason, the expected utilization improvement, and any limitations, and stores them for administrator review.

F. Algorithm

The core logic, in simplified pseudocode, is given below and summarised as a flowchart in Fig. 4.

```
FUNCTION generateRecommendation(date)
  confirmed = valid confirmations for date
  FOR each route R:
    demand[R] = count of confirmed students on R
    free[R] = capacity(bus of R) - demand[R]
  overloaded = routes with free[R] < 0
  proposal = copy of current allocation
  FOR each route O in overloaded:
    excess = -free[O]
    FOR each route C compatible with O:
      IF free[C] > 0 AND excess > 0:
        move = MIN(excess, free[C],
                   eligible students of O
                   at stops shared with C)
        add move to proposal
        free[C] = free[C] - move
        excess = excess - move
    IF excess > 0:
      try an available spare bus;
      IF none, mark O as 'unresolved'
  IF next-morning check fails:
    reject the change
  RETURN proposal, explanation,
       unresolved routes
```

The check on eligible students at shared stops ensures that a student is moved only if the receiving bus can serve that student's approved stop.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue X Oct 2026- Available at www.ijraset.com

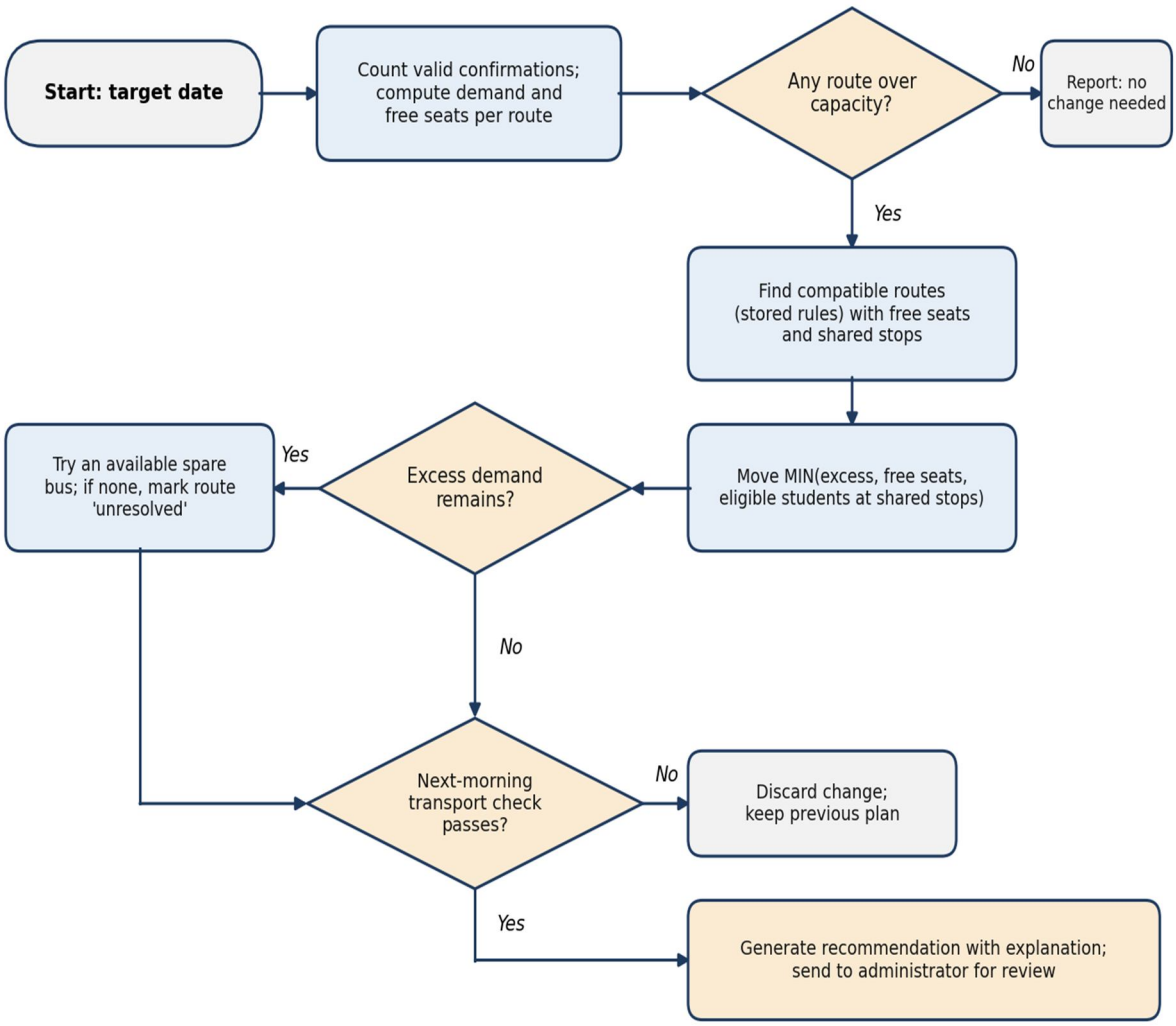


Fig. 4 Flowchart of the rule-based recommendation logic

G. Illustrative Worked Example

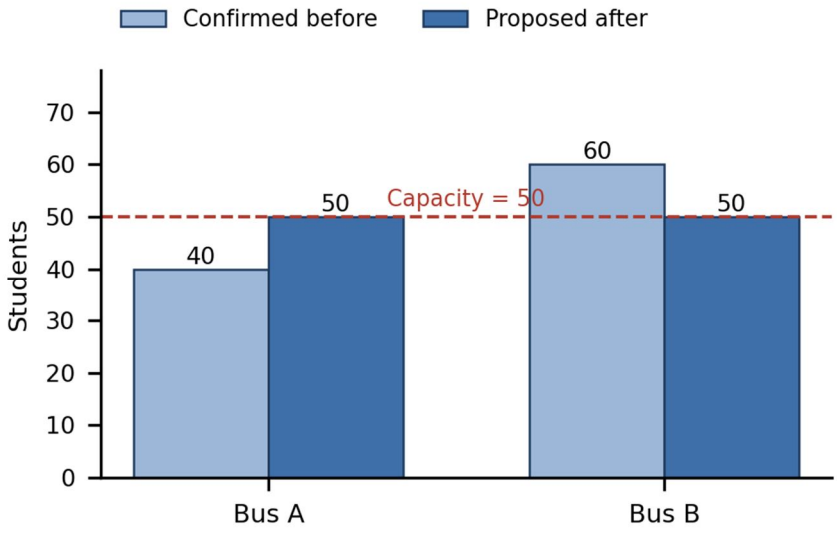
The following example uses assumed numbers to show the logic. It is not measured data. Bus A and Bus B each have 50 seats (Table II).

TABLE II
ILLUSTRATIVE EXAMPLE (ASSUMED VALUES)

Bus	Capacity	Confirmed before	Confirmed after
Bus A	50	40	50
Bus B	50	60 (over capacity)	50

Bus A has 10 free seats before the change and none after. Bus B is 10 seats over capacity before the change and exactly full after. If Route A and Route B are marked compatible and 10 students on Route B board at stops that Bus A serves, the system proposes moving those 10 students to Bus A, and both buses then stay within capacity (Fig. 5). If the routes are not compatible, no transfer is proposed. The system reports the unresolved overload so that the administrator can arrange another solution, such as an additional bus.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



Illustrative example with assumed values (not measured data)

Fig. 5 Illustrative occupancy of Bus A and Bus B before and after the proposed redistribution (assumed values)

V. IMPLEMENTATION

The frontend uses React.js with HTML, CSS and JavaScript, and a responsive layout using Bootstrap or custom CSS. The backend uses Node.js and Express.js and exposes REST APIs for authentication, confirmations, buses, routes, recommendations, approvals and notifications. The agents are implemented as JavaScript modules in the backend. Notifications are created after administrator approval and shown in the student portal. The student confirmation screen and the administrator recommendation review screen of the implemented system are shown in Fig. 6 and Fig. 7.

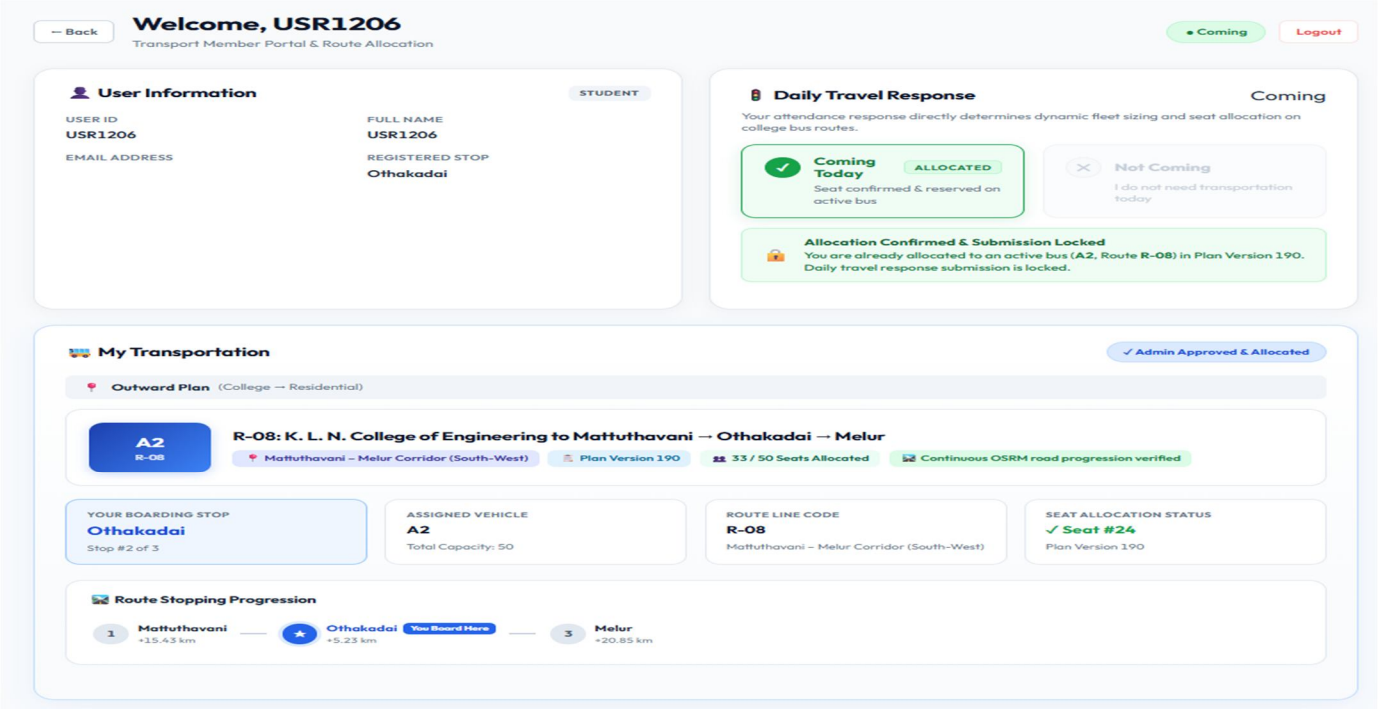


Fig. 6 Student portal: daily travel confirmation screen

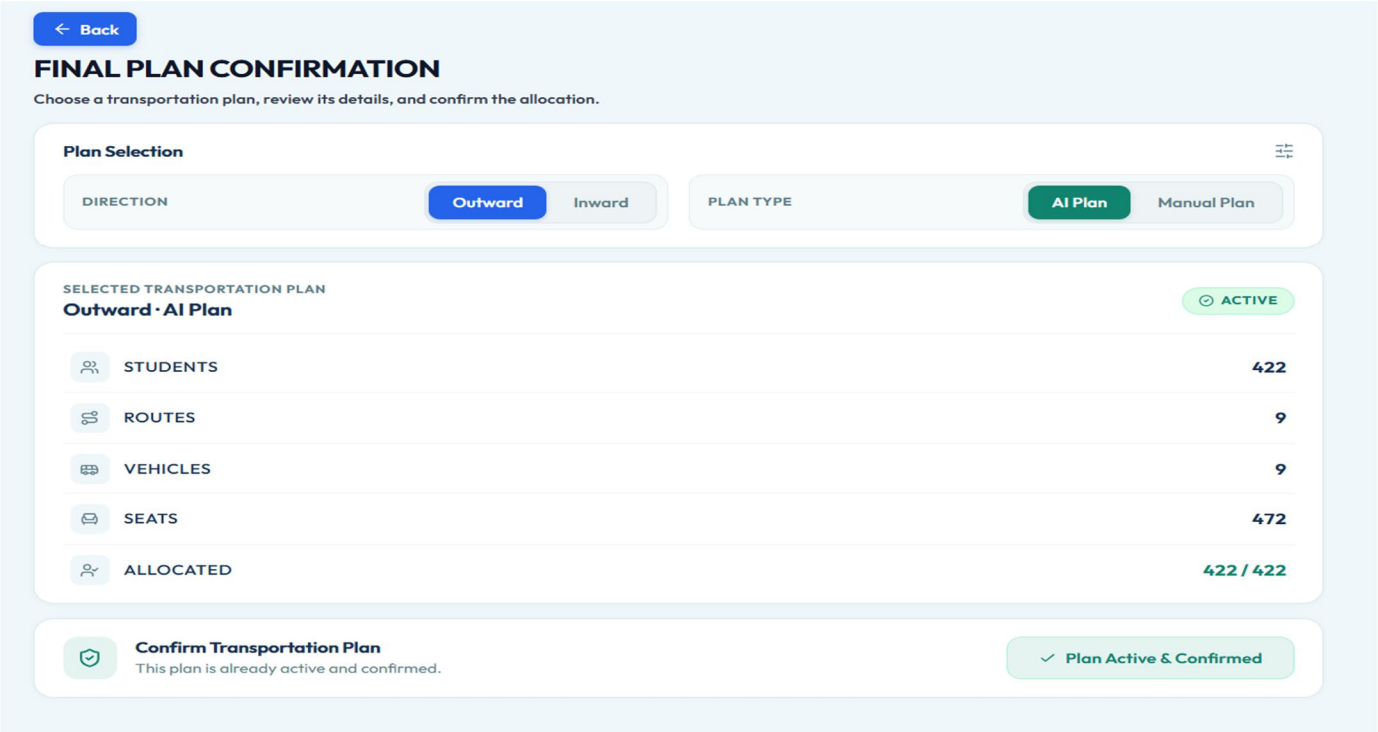


Fig. 7 Administrator dashboard: recommendation review screen

VI. RESULTS AND DISCUSSION

Suggested content, once the system has been tested:

- Functional test results for registration, confirmation, recommendation, approval and notification.
- Test cases for the capacity rule, including a case where no feasible optimization exists and a case with incompatible routes.
- A comparison of seat utilization before and after recommendations, using either real college data (with permission) or clearly labelled simulated data.
- Observations from administrator feedback, if collected.

Discussion points that apply to the design itself are listed below.

- Transparent rules make each recommendation easy to explain and audit.
- Administrator approval keeps responsibility with a human decision-maker.
- The quality of recommendations depends on the quality of the compatibility rules and on how many students submit confirmations.

VII. LIMITATIONS

- The system depends on students confirming on time. Late or missing confirmations reduce accuracy.
- Compatibility rules must be defined and maintained manually by the college.
- It does not predict demand and does not use live location data.
- It has not yet been evaluated at scale on real college data.
-

VIII. CONCLUSION AND FUTURE SCOPE

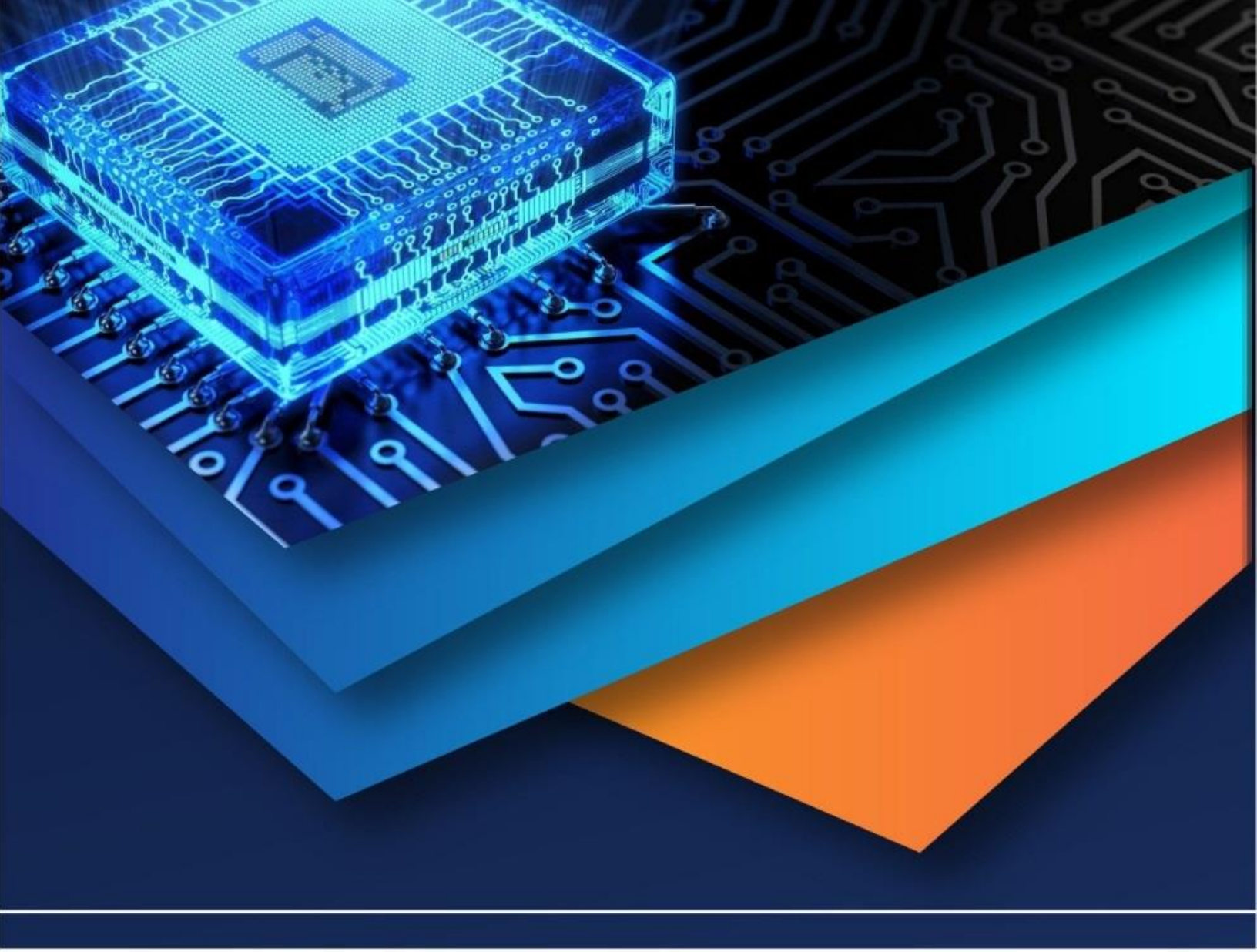
This paper presented a rule-based decision-support system that helps college administrators match bus allocation to actual daily demand. By combining daily confirmations, route compatibility rules, capacity-respecting seat optimization, explainable recommendations, and mandatory administrator approval, the design aims to reduce empty seats while protecting reliability and safety. Possible future enhancements, which are not part of the current system, include SMS or email notifications, reminders for students who have not confirmed, analytics dashboards on historical occupancy, and a pilot study in a real college to measure the effect on seat utilization.

©IJRASET: All Rights are Reserved

REFERENCES

[1] Puneeth Y M and Mithun Dolthody Jayaprakash, "Transportation route optimization," Int. J. Res. Appl. Sci. Eng. Technol. (IJRASET), vol. 13, no. VI, Jun. 2025, doi: 10.22214/ijraset.2025.72760.
[2] G. B. Dantzig and J. H. Ramser, "The truck dispatching problem," Manage. Sci., vol. 6, no. 1, pp. 80–91, 1959.
[3] P. Toth and D. Vigo, Eds., Vehicle Routing: Problems, Methods, and Applications, 2nd ed. Philadelphia, PA, USA: SIAM, 2014.
[4] J. Park and B.-I. Kim, "The school bus routing problem: A review," Eur. J. Oper. Res., vol. 202, no. 2, pp. 311–319, 2010.
[5] M. Jones, J. Bradley, and N. Sakimura, "JSON Web Token (JWT)," Internet Eng. Task Force, RFC 7519, May 2015.
[6] Y. Cai and S. Sun, "Optimizing customized bus routing and maximum seat occupancy rate under the influence of epidemic outbreaks," IEEE Access, vol. 12, 2024, doi: 10.1109/ACCESS.2024.3481427.
[7] S. Parthasarathy, R. Thangasankaran, Gopinath S. B., Harshath K. S., Dinesh Kumar V. V., Jayabalajivel R., and Ranjith Kumar K. B., "TRANSPECTRA: An AI-powered centralised transport management system," Int. J. Comput. Organ. Trends (IJCOT), vol. 16, no. 1, pp. 8–17, 2026.
[8] Y. Harrath, J. AlYusuf, Z. Ghulam, and M. AlDoseri, "Optimal algorithm for managing on-campus student transportation," in Proc. Int. Conf. Decis. Aid Sci. Appl. (DASA), IEEE, 2024, pp. 1–8, doi: 10.1109/DASA63652.2024.10836580.
[9] W. Shu and Y. Li, "A novel demand-responsive customized bus based on improved ant colony optimization and clustering algorithms," IEEE Trans. Intell. Transp. Syst., vol. 24, no. 8, pp. 8492–8506, Aug. 2023, doi: 10.1109/TITS.2022.3145655.
[10] L. Qian and E. Melachrinoudis, "An integrated neural combinatorial tabu search for optimizing school bus scheduling with bell time," Transp. Res. C, Emerg. Technol., vol. 164, Art. no. 104662, 2024, doi: 10.1016/j.trc.2024.104662.
[11] W. A. Ellegood, S. Solomon, J. North, and J. F. Campbell, "School bus routing problem: Contemporary trends and research directions," Omega, vol. 95, Art. no. 102056, 2020, doi: 10.1016/j.omega.2019.03.014.
[12] M. M. Solomon, "Algorithms for the vehicle routing and scheduling problems with time window constraints," Oper. Res., vol. 35, no. 2, pp. 254–265, 1987, doi: 10.1287/opre.35.2.254.
[13] S. Røpke and D. Pisinger, "An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows," Transp. Sci., vol. 40, no. 4, pp. 455–472, 2006, doi: 10.1287/trsc.1050.0135.
[14] D. Huang, Y. Gu, S. Wang, Z. Liu, and W. Zhang, "A two-phase optimization model for the demand-responsive customized bus network design," Transp. Res. C, Emerg. Technol., vol. 111, pp. 1–21, 2020, doi: 10.1016/j.trc.2019.12.004.
[15] P. Vansteenwegen, L. Melis, D. Aktaş, B. D. Galarza Montenegro, F. S. Vieira, and K. Sörensen, "A survey on demand-responsive public bus systems," Transp. Res. C, Emerg. Technol., vol. 137, Art. no. 103573, 2022, doi: 10.1016/j.trc.2022.103573.
[16] Q. Sun, S. Chien, D. Hu, G. Chen, and R. S. Jiang, "Optimizing multi-terminal customized bus service with mixed fleet," IEEE Access, vol. 8, pp. 156456–156469, 2020, doi: 10.1109/ACCESS.2020.3018883.
[17] X. Dou, Q. Meng, and K. Liu, "Customized bus service design for uncertain commuting travel demand," Transportmetrica A, Transp. Sci., vol. 17, no. 4, pp. 1405–1430, 2021, doi: 10.1080/23249935.2020.1864509.
[18] X. Liu, Y.-L. Chen, L. Y. Por, and C. S. Ku, "A systematic literature review of vehicle routing problems with time windows," Sustainability, vol. 15, no. 15, Art. no. 12004, 2023, doi: 10.3390/su151512004.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)