



SMART TRANSPORT SYSTEM

Kate P. S.¹, Kurkute Shruti²

¹ Faculty Guide & Supervisor | ² Dattakala Group of Institutions, Faculty of Engineering, Pune, Maharashtra, India, India

Corresponding author: shrutikurkute6@gmail.com

ARTICLE ID IJREK-2026-00159	ARTICLE TYPE Research Article	RESEARCH AREA Machine Learning & Deep Learning	PUBLICATION Volume 1, Issue Issue 1 (2026)
RECEIVED 25 October 2026	REVISED 27 October 2026	ACCEPTED 26 October 2026	PUBLISHED 28 October 2026

Abstract

Urban and semi-urban transport systems face challenges such as traffic congestion, delays, inefficient route planning, limited real-time information, and difficulty coordinating transport services. This paper proposes a Smart Transport System that uses digital technologies to improve access to transport information and support more efficient travel planning. The proposed system may combine route information, vehicle or service updates, location-aware features, and a user interface that presents useful travel details. Depending on available data sources, it can help users view routes, estimated arrival information, and service notifications. The system is intended as a decision-support application and its reliability depends on data freshness, connectivity, and the accuracy of the underlying transport information. A prototype may be evaluated for response time, route-information accuracy, usability, and availability. No experimental performance results are claimed because the proposed system has not yet been benchmarked. The design emphasises data protection, accessibility, and clear communication when real-time information is unavailable.

Keywords: Smart Transport; Intelligent Transportation System; Route Planning; Real-Time Information; Mobility; Transport Management

1.1 PROBLEM STATEMENT

The problem is to design an accessible transport-information system that helps users view routes and service details, plan journeys, and receive relevant updates while handling inaccurate, incomplete, or unavailable data transparently. 1.2 Objectives • Provide a simple interface for accessing transport and route information. • Support journey planning using available route and location data. • Display service updates or estimated arrival information when a reliable data source exists. • Improve usability through clear navigation and accessible presentation. • Evaluate information accuracy, response time, usability, and reliability. 1.3 Scope The proposed project is an academic prototype. Its exact functions depend on the selected transport mode and available data sources. It does not control vehicles or guarantee journey times, and it should clearly label estimates and data timestamps.

2. LITERATURE REVIEW

Traditional transport information services commonly provide timetables, route maps, and service notices. While useful, separate systems may make it difficult for travellers to compare options or respond quickly to disruptions. Intelligent Transportation Systems (ITS) apply information and communication technologies to transport operations, traveller information, and traffic management. Digital maps and location-based services can support route discovery and navigation. Real-time vehicle feeds, where available, can provide updates about vehicle locations, arrivals, or service status. Data integration is a central challenge: different operators may use different formats, update frequencies, and definitions. Missing or stale data can reduce the usefulness of a displayed estimate. A practical smart transport system should therefore combine a clear user interface with transparent data provenance, appropriate error handling, and evaluation using representative journeys. It should avoid implying that a service is live unless the underlying data source actually provides live updates.

2.1 RESEARCH GAP

A transport application must do more than display a route: it should present available information consistently, indicate when information was last updated, and

handle disruptions or missing data gracefully. The proposed design addresses these needs through modular data handling, clear status messages, and user-focused journey information.

3. PROPOSED SYSTEM

The proposed system allows a user to select a starting point and destination, view available transport or route details, and inspect relevant service information. If connected to a suitable provider or operator feed, the system may show vehicle locations, arrival estimates, or disruption notifications. Otherwise, it should display timetable or static route information with a clear label. The application may also include saved routes and a basic journey history if these features are within project scope.

Figure 1. Proposed processing flow (conceptual): User Origin and Destination → Data Retrieval → Route/Service Processing → Data Freshness Check → Journey Results and Updates

3.1 MAIN MODULES

TABLE I

Module	Purpose
User interface	Collect origin, destination, and user preferences.
Transport data integration	Retrieve route, timetable, and service details from available sources.
Route processing	Organise available journey options and relevant transport information.
Update handling	Show arrival estimates or disruptions only when supported by data feeds.
Notifications and report	Present useful updates, status timestamps, and clear fallback messages.

3.2 SYSTEM ARCHITECTURE

A prototype may use a web or mobile interface with a backend API that validates requests and retrieves transport information. A database can store configuration, saved routes, and limited user preferences where necessary. Map and route services may be

integrated subject to their terms and availability. If a public transport feed is used, the implementation should document its coverage and update frequency. Authentication, access controls, and secure handling of API keys should be included where relevant. 4. Methodology

4.1 REQUIREMENTS AND DATA SOURCES

First, define the target transport mode and user needs, such as bus routes, public transport timetables, vehicle tracking, or multi-modal journey planning. Identify reliable data sources and confirm their coverage, licence conditions, update frequency, and permitted usage. Where live data is unavailable, the prototype should use clearly labelled static or sample data rather than suggesting that the information is real-time.

4.2 DATA PROCESSING

• Validate origin, destination, and other user inputs. • Check transport data format, completeness, and update timestamp. • Standardise route names, stop identifiers, and time formats where applicable. • Handle missing, delayed, or inconsistent records with understandable messages. • Avoid exposing unnecessary location history or personal information. 4.3 Route and Service Information The system can retrieve or organise available route information and present suitable options according to the chosen project scope. If a routing service is used, the application should respect provider limits and report failures. Arrival estimates should be shown only when supported by an appropriate source; the system should label estimates as approximate and indicate when the feed was last updated.

4.4 USER NOTIFICATIONS

Notifications may be generated for service disruptions, route changes, or saved-journey updates when the relevant information source supports these events. Notifications should be timely, concise, and configurable. The application should avoid repeatedly notifying users about unchanged information and should provide a way to disable optional alerts.

4.5 CONCEPTUAL ALGORITHM

1. Receive the user's journey request. 2. Validate the origin and destination. 3. Retrieve available transport data. 4. Check data coverage and freshness. 5. Organise route and service information. 6. Label live data, static information, and estimates accurately. 7. Display results or a clear fallback message. 8. Collect usability feedback and evaluate the system.

5. IMPLEMENTATION PLAN

TABLE II		
Layer	Suggested technology	Responsibility
Interface	React or HTML/CSS/JavaScript; mobile app if required	Collect journey inputs and display results.
Backend API	Node.js/Express or FastAPI	Validate requests and coordinate data retrieval.
Maps and routing	A permitted map/routing API or documented open data	Provide map and route details where available.
Data processing	JavaScript/TypeScript or Python	Normalise and check route/service data.
Storage	PostgreSQL, MongoDB, or minimal local storage	Store necessary settings and saved routes.
Testing	Unit, integration, and usability tests	Check reliability, correctness, and user experience.

5.1 SECURITY AND PRIVACY REQUIREMENTS

• Use HTTPS and restrict access to protected user information. • Keep API keys and credentials outside source code. • Request location permission only when needed and explain why it is used. • Minimise storage of precise location history and provide deletion controls where applicable. • Respect map, routing, and transport-data provider terms.

6. EVALUATION PLAN AND DISCUSSION

TABLE III		
Metric		Meaning
Route information correctness	Whether displayed routes and stops match the selected data source.	
Arrival estimate error	Difference between displayed estimates and observed arrivals, if suitable data exists.	
Response time	Time taken to return journey results.	
Availability	Whether the service responds successfully under expected conditions.	
Usability	Whether users can find and interpret journey information easily.	
Data freshness	Age of the latest available transport update.	

7. ADVANTAGES AND LIMITATIONS

7.1 Advantages

• Can bring available journey and route information into one interface. • May reduce uncertainty when reliable service updates are available. • Can support different transport modes as data integrations are added. • Can be extended with saved routes, accessibility information, or disruption alerts. 7.2 Limitations • Live updates depend on the availability and quality of transport data feeds. • Network failures or provider outages may interrupt access to information. • Estimated arrival times can change due to traffic, weather, or service disruptions. • Map and routing services may impose usage limits or licensing requirements. • The system cannot guarantee that a displayed route or arrival estimate will remain accurate.

8. ETHICAL AND RESPONSIBLE USE

A smart transport application should present information clearly and avoid overstating its accuracy. Users should understand whether details are live, scheduled, estimated, or static. Location data should be collected only when needed, protected appropriately, and retained for a limited period. The interface should consider accessibility and provide clear alternatives when information is unavailable. Transport data should be used in accordance with the relevant provider's terms.

9. FUTURE SCOPE

Future work may include multimodal journey planning, accessibility-aware routes, fare information, crowding indicators, integration with public transport feeds, regional-language support, offline access to saved routes, and smarter disruption alerts. These features should be introduced only when dependable data and suitable validation are available. 10. Conclusion This paper proposed a Smart Transport System for presenting route and service information through a unified digital interface. The proposed design includes transport-data integration, route processing, data-freshness checks, and clear communication of estimates and limitations. Its practical value must be established through testing with representative journeys and dependable data sources. The system should assist travel planning rather than promise guaranteed journey times.

REFERENCES

[1] International Organization for Standardization. (2015). ISO 14813-1:2015 Intelligent transport systems — Reference model architecture(s) for the ITS sector — Part 1: ITS service domains, service groups and services. ISO.

[2] U.S. Department of Transportation, Intelligent Transportation Systems Joint Program Office. Intelligent Transportation Systems (ITS) resources and programme materials.

[3] Google Maps Platform. Routes API documentation. Documentation for route computation and related services; consult current terms and coverage before use.

[4] General Transit Feed Specification (GTFS). Schedule and real-time transit data specifications.

[5] World Bank. (2021). Urban Mobility and Public Transport: relevant transport planning resources.

[6] National Institute of Standards and Technology. (2023). Artificial Intelligence Risk Management Framework (AI RMF 1.0). NIST AI 100-1. Note: The references provide background and implementation context. Any claims about the prototype's performance should be added only after implementation and evaluation.

Publication archival note: Official scholarly contribution published under CC BY 4.0 International License. Permanent record on <https://www.ijrek.org> | Article ID: IJREK-2026-00159