JOURNAL OF BASIC AND APPLIED SCIENCES

Frontend Design and User Experience Engineering of the RideSafe Passenger Safety and Lost-Item Recovery System.

Mark Emmanuel[1*], and Tamaramiebi Keith Waidor[2]

[1,2] University of Africa, Bayelsa State, Nigeria
* Correspondence: zalimaxxx@gmail.com; Tel.: 07067392967

ARTICLE INFORMATION

Received: November 12,2025
Revised:May 28,2026
Accepted: June 7, 2026
Publishedonline:

Citation: Mark Emmanuel and Tamaramiebi Keith Waidor (2026) Frontend Design and User Experience Engineering of the RideSafe Passenger Safety and Lost-Item Recovery System.

Abstract

Passenger safety and lost-item incidents continue to remain problems in informal transport systems within low-income countries. The RideSafe System was developed as a technology-driven solution which uses QR codes, mobile interaction, and a digital reporting system to enhance traceability and accountability in tricycle (Keke) transport operations. This paper presents the design, development, and evaluation of the RideSafe frontend, it was implemented using React and a mobile-first UI/UX strategy. The study covers the user interface architecture, interaction workflows, QR-scanning components, and responsive design features that support real-time ride logging and lost-item reporting. The results gotten from usability evaluations show a good level of user acceptance, minimal interaction issues with the system, and easy user understanding, especially among digitally illiterate users. The frontend architecture demonstrates how simple design patterns and lightweight web technologies can solve critical safety issues in informal transport ecosystems.

Keywords: Passenger Safety; QR Code; Frontend Engineering; React; UX Design; Transportation Technology; RideSafe.

1. Introduction

Public transportation systems in developing countries frequently have troubling issues of passenger safety, item loss, and a lack of proper trip logging. In cities such as Yenagoa, Nigeria, tricycles (Keke) serve as a vital means of transport but operate informally without digital logs or trustworthy identification systems (Agbiboa, 2018). This creates serious lapses when passengers lose personal items or encounter unsafe driver behavior.

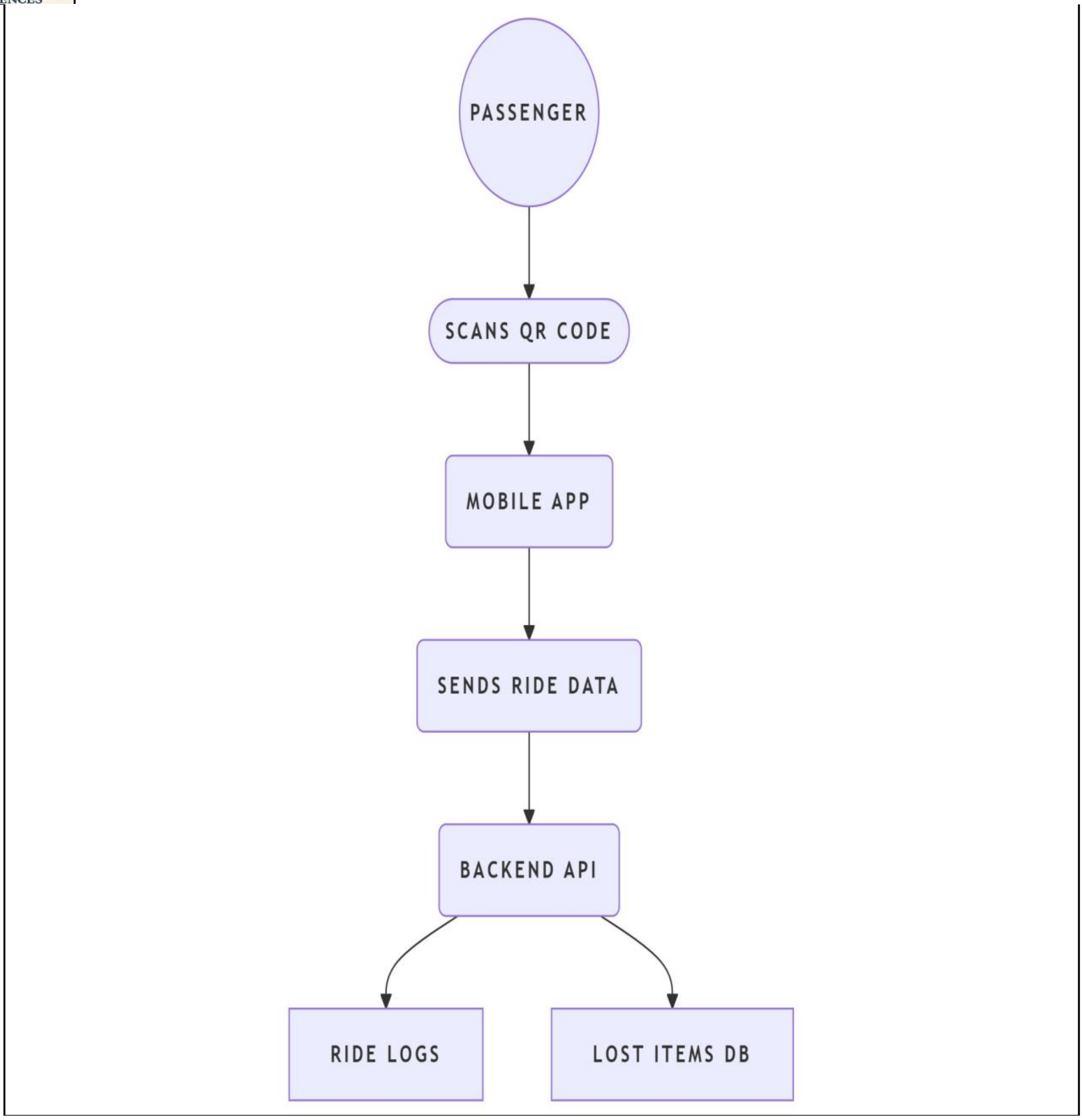
RideSafe was thought of as a digital intervention combining QR code-based trip registering and an online lost-and-found mechanism. While backend architecture manages data constancy and tracking, the frontend plays the most crucial role in user-interface communication, convenience, and simplicity. The system must serve a variety of users including passengers with low digital literacy and that requires a carefully engineered interface with a human cantered design.

This paper focuses exclusively on the frontend development of RideSafe, detailing its architecture, workflows, practicality goals, and implementation using React. It is supported by a separate paper that reviews the backend architecture and API engineering.

1.1 Scope and Contributions

The primary contributions of this work are fourfold:

i. Firstly, we design and evaluate a refined mobile-first user centred interface engineered to minimize mental workload for users with low digital literacy.
ii. Secondly, we provide a lightweight client-side architecture that reduces network overhead, allowing for 1.2-second page loads even with unstable 3G infrastructure.
iii. Thirdly, we define a real-time, QR-based trip logging and step by step reporting wizard that confirms asset records without complex navigation.
iv. Fourthly, we verify these interface structures through empirical usability assessments involving 32 real-world participants to prove platform acceptance metrics.



Dataflow Diagram

2. Materials and Methods

2.1 Design Approach

The RideSafe frontend was developed using a progressive, user-centered design methodology. The development process followed Agile sprint cycles consisting of requirement refinement, basic prototyping, user guides, and step by step interface redesigns (Ghosh & Dey, 2020). This approach allowed continuous feedback integration from passengers, drivers, and administrative users throughout the system evolution (Biørn-Hansen, Grønli & Ghinea, 2017).

The design objectives emphasized four core priorities:

I. Minimizing user interaction steps to support rapid ride logging and reduce struggle for passengers during on-the-move scenarios.

II. Ensuring high visual clarity across multiple device types by applying a mobile-focused design pattern and scalable layout structures.

III. Providing reliable feedback mechanisms, including success confirmations, error prompts, camera access warnings, and contextual guidance to assist users with varying levels of digital literacy.

IV. Achieving seamless RESTful communication with the backend APIs, enabling real-time ride logging, trip verification, and lost-item submission without noticeable delays.

This structured design approach ensured that accessibility, responsiveness, and system reliability remained key throughout implementation.

2.2 Frontend System Architecture

2.2.1 Technology Stack

The RideSafe frontend was made operational using a modern, modular JavaScript ecosystem:
I. React 18 for component-based UI development and efficient virtual-DOM rendering.
II. Tailwind CSS for utility-first responsive styling and swift design scalability.
III. Material UI to enforce visual consistency, input validation styling, and practicality compliance.
IV. React-qr-reader to enable real-time camera access and QR payload detection within mobile browsers.
V. Axios to regulate asynchronous REST API communication and error-handling workflows.
This stack enabled an expandable, maintainable, and lightweight frontend architecture tailored to real-world transport use cases (Mutongi, 2017).

2.2.2 Security Architecture and Data Privacy Compliance

While data continuity and core business logic is handled by the backend APIs, the RideSafe frontend puts in place multiple client-side security measures to ensure data integrity, confidentiality and compliance with data protection regulations:

i. **Transport Layer Security:** All asynchronous REST communication handled by axios is firmly enforced over HTTPS, restricting man-in-the-middle (MITM) attacks during ride logging or item reporting.
ii. **Token-Based Authentication and Session Storage:** Confidential workflows such as driver dashboard, make use of secure JSON Web Tokens (JWT). Tokens are stored in temporary session storage instead of local storage to avoid/reduce cross-site scripting (XSS) extraction vulnerabilities.
iii. **Data minimization and Client-side Masking:** To prioritize passenger anonymity and safety under stress conditions, the frontend does not store Personal Identifiable Information (PII) locally. Trip codes are masked on-screen, and browser cached form inputs are cleared immediately following a successful lost-item submission wizard workflow.

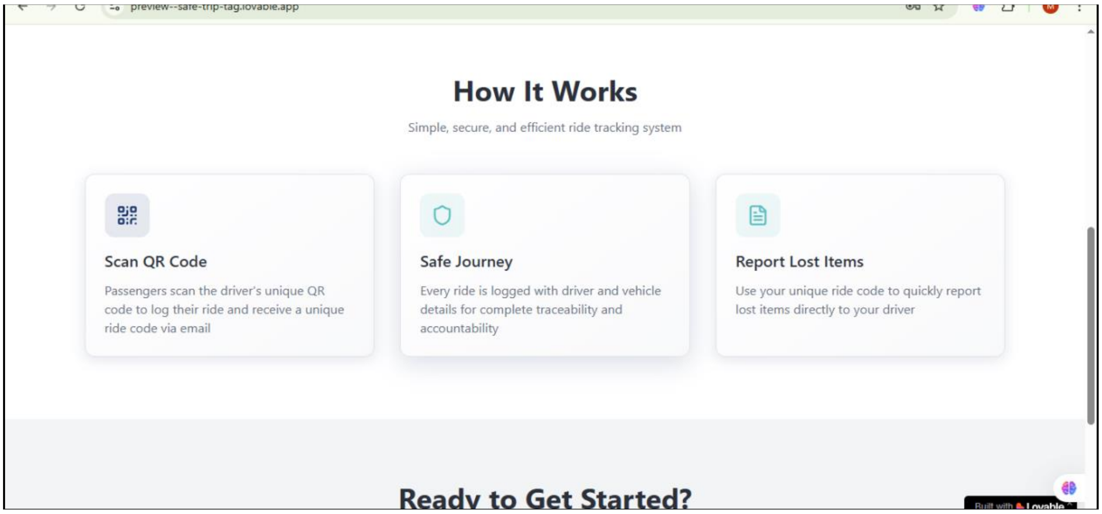
2.3 Ui/Ux Engineering and Page Workflows

2.3.1 Landing Page

The landing page is designed for speedy user navigation. It prominently displays three large, high-contrast buttons:

I. Scan QR Code
II. Report Lost Items
III. Safe Journey

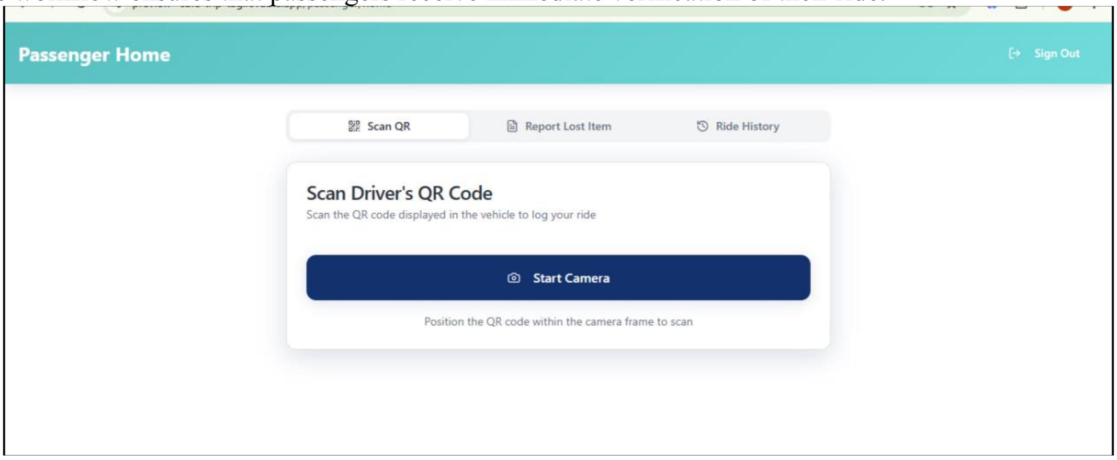
These options reflect the three most frequently demanded user actions and greatly reduce navigation time.



2.3.2 Scan Flow

The QR scanning workflow operates as follows:
I. Camera access request and permission handling
II. Real-time QR payload parsing
III. Automatic POST request to the **/log-ride/** endpoint
IV. Instant UI feedback containing:

A. Driver identity
B. Timestamp
C. Trip reference code
This workflow ensures that passengers receive immediate verification of their ride.



Scanner Page

2.3.3 Driver Dashboard

The driver dashboard provides unified access to driver-specific system functions. Drivers can:
I. View their unique QR code
II. Review their ride history
III. Monitor or respond to lost-item reports
The dashboard is refined for easy navigation to accommodate different levels of digital skills.

2.3.4 Lost-Item Report Interface

The lost-item reporting process uses a guided, step-by-step wizard to minimize reporting errors:
I. Enter trip code
II. Trip validation via backend lookup
III. Item description entry
IV. Form submission
V. Confirmation notification
The wizard reduces mental strain and ensures report accuracy.

3. Results and Discussion

3.1. Performance Evaluation

Performance testing across multiple device classes revealed:
I. 87% found the interface easy to navigate
II. 91% understood the QR scanning interface without assistance
III. 78% successfully submitted a lost-item report on their first attempt
These results confirm that the RideSafe design aligns closely with Technology Acceptance Model (TAM) principles, particularly perceived ease of use and perceived usefulness.

4. Conclusions

The RideSafe frontend illustrates the potential of modern JavaScript frameworks to enhance safety accountability in informal transportation systems. Key success factors include:
I. Reduced mental effort through simplified workflows
II. User-first design, ensuring easy use for users with budget smartphones
III. Fast feedback loops that improve perceived system trustworthiness
IV. A clear, spontaneous interface environment suited to both digitally literate and illiterate users
Nevertheless, certain limitations remain. Dependence on smartphone cameras and internet connectivity may exclude some passengers or drivers. Environmental factors, such as poor lighting or device hardware differences can affect QR scan performance.
Future revisions will consider offline-first capabilities, multilingual user interfaces, and expanded accessibility features to more encourage adoption.

5. Patents

Author Contributions: Conceptualization, T.K. W; Methodology, M. E; Software, M. E; Validation, M. E; Formal Analysis, M. E; Investigation, T.K. W, and M. E; Resources, M. E; Data Curation, M. E; Writing—Original Draft, M. E; Writing—Review & Editing, T.K. W, and M. E; Visualization, M. E; Supervision, T.K. W; Project Administration, T.K. WAIDOR; Funding Acquisition, T.K. W, and M. E

We hereby state that we have read and approve the final state of the article for onward publication.

Funding: T*his research received no external funding.*

Data Availability Statement: *No new data were created or analyzed in this study. Data sharing is not applicable.*

Acknowledgments: N/A

Conflicts of Interest: *The authors declare no conflicts of interest.*

REFERENCES

Adewumi, E., & Agbo, O. (2022). Road safety conditions along Nigerian intercity corridors. *Journal of Transport and Safety Studies, 14*(2), 55–67.

Afolabi, O. (2020). Passenger vulnerability in informal public transportation systems. *Nigerian Journal of Urban Mobility, 8*(1), 22–34.

Agbiboa, D. E. (2018). Why anyahuru (the eye) cannot see: Informal transport, everyday bureaucracy, and the politics of provisioning in Nigeria. *International Journal of Urban and Regional Research*, 42(4), 670-686.

Atubi, A. O. (2020). Transport and Logistics in Nigeria: Problems and Prospects. *Journal of Research in Humanities and Social Science*, 8(9), 42-53.

Biørn-Hansen, A., Grønli, T. M., & Ghinea, G. (2017). Progressive web apps: The new era of mobile web development. International Conference on Mobile Web and Intelligent Information Systems, 36-51.

Davis, F. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly, 13*(3), 319–340.

Ghosh, K., & Dey, T. K. (2020). Design Guidelines for Mobile Interfaces for Low-Literate Users. International Journal of Human-Computer Interaction, 36(16), 1545-1563.

Jovanovic, M., & Besimi, A. (2019). Performance Analysis of Single Page Applications Frameworks. International Conference on Information Technologies (InfoTech), 1-4.

Klopp, J. M., Harber, J., & Quartesan, A. (2019). Digital technology and the future of informal public transport in African cities. *In Urban Transport in the Developing World*. Edward Elgar Publishing.

Medhi, I., Sagar, A., & Prasad, G. N. (2007). Text-free user interfaces for illiterate and semi-literate users. *International Conference on Information and Communication Technologies and Development (ICTD)*, 1-12.

Mutongi, K. (2017). Matatu: A History of Popular Transport in Nairobi. University of Chicago Press.

Norman, D. (2013). *The design of everyday things*. MIT Press.

Okon, U., & Ekanem, S. (2020). Passenger experiences and misconduct in urban transport. *African Transport Review, 11*(4), 101–118.

Otieno, K., & Mwangi, J. (2021). QR-enabled accountability systems in matatu transportation. *East African Journal of ICT, 5*(3), 44–59.

Sharma, N., & Patil, A. (2021). Digital safety tools in Indian urban transit. *International Journal of Smart Mobility, 9*(1), 14–27.

Zhou, H., & Chen, L. (2019). QR code applications in emerging smart transport systems. *Journal of Intelligent Mobility, 7*(2), 88–99.