IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 5, November 2025
Impact Factor: 7.67

QR Code Based Bus Pass Authentication System

Sushmitha K M and Raghavendra G N
Department of MCA
Vidya Vikas Institute of Engineering and Technology, Mysuru, India
sushmithakm159@gmail.com

Abstract: *Public transport remains the backbone of urban mobility in India, yet routine ticketing and pass verification are still dominated by paper-based processes. Paper passes are easy to lose, hard to verify at scale, and vulnerable to misuse, while manual checks slow down boarding and reduce operational transparency. This project presents a QR Code Based Bus Pass Management System, an Android application developed in Java (Android Studio, Gradle Groovy) that digitises the entire pass lifecycle. The solution supports three roles— Admin, Passenger, and Conductor—within a single app. Passengers purchase and store digital passes locally and present them as QR codes; Conductors validate these QR codes using CameraX and Google ML Kit (Barcode Scanning); Admins manage users and pass products. Data is persisted offline using Room (SQLite), while SharedPreferences supports role-aware session management. The interface follows Material Design 3, providing a clean, accessible, and responsive user experience suitable for devices commonly used in Indian cities.*

The system is designed to work reliably even when data connectivity is poor, a practical requirement on many Indian routes. Room Database tables—users, passes, and shift_logs—enable consistent local operations: passengers can view active passes and history, conductors can scan and log validations during shifts, and admins can oversee user administration and pass definitions from the same APK. The design also includes placeholders for future enhancements such as password hashing, key rotation, and optional server synchronisation, ensuring a pathway to production hardening without compromising today's usability.

Keywords*:* missing person identification, deep learning, face recognition, data augmentation, Flask, MySQL, real-time video analytics, geolocation alerts

I. INTRODUCTION

Public bus networks in India transport lakhs of passengers daily across cities and peri-urban areas. While e-commerce and digital payments have seen rapid adoption, the everyday experience of buying and carrying a monthly pass often remains analogue: queues at depots, paper cards with stamps, and manual inspections by conductors. This analogue workflow is time-consuming for commuters and operationally inefficient for transport authorities. It also lacks strong safeguards against duplication or misuse. In this context, a mobile solution that digitises passes, streamlines validation, and works without constant internet becomes both relevant and feasible.This project introduces a QR Code Based Bus Pass Management System, an Android application that consolidates three roles—Admin, Passenger, and Conductor—in a single codebase. The Passenger registers, buys a pass that is stored locally, views an on-device QR code for the active pass, and can review purchase history. The Conductor uses the app to scan a passenger's QR code and validate its status instantly, logging each successful check to a Shift Log for later review. The Admin role manages users and pass products and has a reserved default credential for initial setup. This role-based design reflects the day-to-day operations of Indian bus services, where multiple responsibilities must be handled by different people on the move.Technically, the app is built in Java with Android Studio and Gradle (Groovy). It uses Room (SQLite) to ensure offline data persistence, which is essential for routes with patchy connectivity. SharedPreferences implements simple, reliable session handling and role-aware redirects after login. CameraX paired with Google ML Kit (Barcode Scanning) provides fast, on-device QR scanning without external dependencies. The UI adopts Material Design 3 guidelines with card- based layouts, gradient backgrounds, and forms using Material components to keep the experience familiar and

DOI: 10.48175/IJARSCT-30011

accessible.The system balances practicality and forward compatibility. For demonstration, credentials are stored plainly, yet the codebase already marks a path for password hashing and key rotation to strengthen security. Similarly, while the current version operates fully offline to fit Indian field conditions, the data layer and DAOs are designed so that a future sync module or remote API can be added without rewriting core flows.

II. PROBLEM STATEMENT

Traditional bus pass workflows rely on paper cards and manual verification, leading to delays, loss or wear-and-tear of passes, and susceptibility to misuse or duplication. Conductors must visually inspect passes, which is error-prone in crowded conditions and offers limited traceability. Existing digital attempts often assume constant connectivity or require separate apps for different roles, complicating operations. There is a need for a single, offline-capable Android solution that enables passengers to present a secure, scannable digital pass, allows conductors to validate quickly with a QR scan, and equips administrators to manage users and pass products with minimal training and infrastructure.

III. LITERATURE SURVEY

Early work on QR-code–enabled ticketing in public transport focused on the smart city vision and high- level architecture. Fong et al. review smart bus applications that use QR codes for ticket purchase and validation and compare them with alternatives such as NFC and smartcards, arguing that QR offers a lower infrastructure cost and can run on commodity smartphones without dedicated validators [1]. Their review highlights key building blocks—mobile client, backend code generation, and on-board validation—and emphasises real- world challenges such as scanning under poor lighting, operator process integration, and the need for secure code formats.ResearchGate+2Semantic Scholar+2 This provides a conceptual foundation for QR-based bus pass systems that target cost- sensitive agencies like Indian city operators.

A second strand of work focuses specifically on student bus passes. Swetha et al. propose a QR- based student bus pass system in which students register online, receive a digital pass encoded as a QR code, and present it to conductors for validation via an Android application [2].IJARST+2IJARST+2 Their design replaces paper passes and manual stamping with digital issuance and renewal, reducing queues and improving data consistency for educational institutions and operators. The system records essential metadata—student details, route information, and validity—creating a cleaner audit trail than manual registers, but it still assumes some level of online connectivity for issuance and database updates.

Other researchers broaden the focus from passes to journey-level e-ticketing. Karale et al. develop an

E-Ticket Booking System for Public Transportation using QR Code ‖ where passengers generate QR-coded tickets and conductors validate them during the journey; each scan writes journey details to a backend, thus logging validation events for reconciliation and analysis [3].IOSR EECS+3IJARSCT+3ResearchGate+3 Their work underlines how real-time capture of scan events can strengthen transparency and simplify settlement, but the architecture is more online-centric, with database updates and reporting assumed to be network-connected during operation.

A closely related line of work looks at end-to-end online bus pass lifecycle management. Krishnammal et al. present an ―Efficient Bus Pass Generation and Authentication using QR Code ‖ framework in which registration, pass issue, renewal, and authentication are moved entirely online, with distinct roles for user and admin and a unique pass identifier encoded in the QR [4].ijitee.org+2ResearchGate+2 The system reduces paperwork and speeds up checks, and its data entities (user, pass, validity, ID in QR) have become a template for later Android implementations. However, it presumes a server-backed model and does not specifically address offline operation on routes with poor connectivity.

More recent contributions emphasise fully Android-based implementations and richer user flows. Vaidianathan describes an Android smart bus pass application with separate logins for users and administrators, where students register, receive OTP-based verification via email, and upon admin approval obtain digital passes that can be renewed and stored on the device [5].grnjournal.us+3ResearchGate+3grnjournal.us+3 The system includes features such as email delivery of passes and renewal reminders, framing the app as a practical blueprint for student and general commuters. Yadav et al. go a step further by framing a ―modernized ‖ smart bus pass as an integrated application

DOI: 10.48175/IJARSCT-30011

combining registration, a digital wallet or payment interface, QR generation, and admin dashboards in a single modular design [6].Scribd+3ijsrem.com+3ijsrem.com+3 Their article stresses that consolidating these modules into one mobile experience reduces friction and supports scale, while also acknowledging constraints like intermittent connectivity and device diversity.

Beyond pure software, some work explores payment flexibility and financial workflows within QR-based ticketing. Gaikwad et al. propose Smart Bus, a smartphone-based framework where QR codes are used for ticketing alongside a ―delay payment‖ method [7].IJRASET+2IOSR EECS+2 Their design supports both standard online payments and deferred settlement, recognising that immediate wallet deduction is not always practical in real-world operations. Each QR scan authenticates the fare and can log trip data for operators, illustrating how QR ticketing can adapt to the cashflow and network realities of smaller operators.

Tupare et al. focus on online payment integration with QR codes in ―Smart Public Transport Ticketing System Using QR Code Online Payment Method,‖ where an embedded or kiosk-like stack (microcontroller, keypad, TFT display) collaborates with a backend (e.g., Flask API) to generate payment links rendered as QR codes for passengers to scan and pay [8].ijsrd.com+3ResearchGate+3SSRN+3 Their prototype demonstrates that QR-based ticketing is not limited to smartphones; instead, inexpensive hardware can host the purchase and validation surface at terminals or on vehicles, closing the loop between fare selection, payment, and confirmation in real time.

Taken together, these systems show a clear evolution: from conceptual smart-city bus applications using QR codes for general ticketing [1], through student-specific and web-backed bus pass solutions [2], [4], to Android-centric implementations with integrated registration, approval, and digital wallets [5]–[8]. However, several gaps remain for contexts such as Indian urban and peri-urban bus networks:

- Most implementations either assume reliable connectivity for server interaction or treat offline mode as secondary; they do not deeply exploit local databases like Room/SQLite for fully offline operations.
- Roles (Admin, Passenger, Conductor) are often split across distinct systems or apps, rather than consolidated into a single APK with clean role segregation and shared data models.
- While logging is discussed (e.g., journey logs or basic metadata), few works provide a structured Shift Log model that can be locally persisted and later synced for audit and analytics.
- Security discussions typically stop at unique IDs in QR codes, without explicit design for future hardening (password hashing, key rotation, cryptographic signatures) within an offline-first architecture.

The proposed QR Code Based Bus Pass Management System is positioned to address these gaps by (i) packaging Admin, Passenger, and Conductor roles into one Android application, (ii) relying on Room (SQLite) and SharedPreferences to support robust offline operation and role-aware sessions, and (iii) using CameraX + Google ML Kit for on-device QR scanning while maintaining a tamper-resistant Shift Log that can be synced to a backend at a later stage. This combination of offline reliability, multi-role support in a single APK, and explicit provision for future security and synchronisation extends the capabilities reported in existing literature and targets the specific operating conditions of Indian bus services.

IV. METHODOLOGY

The proposed QR Code Based Bus Pass Management System was engineered as a single- APK, role-based Android application using Java in Android Studio with Gradle (Groovy) as the build system. The development began with translating the functional requirements—three roles, offline operation, QR-based validation, and audit logs— into a layered architecture comprising presentation, domain, and data layers. The presentation layer uses Activities/Fragments and ViewModels with Material Design 3 components to ensure a consistent UI for Admin, Passenger, and Conductor. The domain layer encapsulates business logic for authentication, pass lifecycle management, QR code handling, and shift logging so that UI code remains thin and testable. The data layer relies on Room (SQLite) for persistent storage and SharedPreferences for lightweight session management. Entities such as User, PassProduct, Pass, and ShiftLog are modelled as Room entities with corresponding DAOs to support type-safe, compile-time checked database operations.

DOI: 10.48175/IJARSCT-30011

This separation of concerns was chosen to keep the app maintainable and to provide clear extension points for future server synchronisation and security upgrades.

From a workflow perspective, the user and pass management logic was designed around explicit role separation while still sharing one codebase. On first launch, a default Admin credential is seeded in the local database to bootstrap the system. Admin users log in and access an Admin Dashboard, where they can create and manage user accounts (assigning roles Admin/Passenger/Conductor) and configure pass products (e.g., monthly, weekly, student passes) by specifying duration, price (demo), and status. Passenger registration is either self-service or admin-driven depending on deployment. After login, the system writes the authenticated user's ID and role to SharedPreferences; on subsequent app launches, this stored session is used to route the user directly to the appropriate dashboard. This role-aware navigation ensures that each user only sees actions relevant to their duties while internally using the same Activities/Fragments where feasible to reduce duplication.

The pass lifecycle and QR code handling form the core functional pipeline. When a Passenger purchases a pass, the app creates a new Pass row in Room with a unique pass identifier (e.g., UUID), references to the passenger and chosen pass product, and computed start and expiry dates based on the product's duration. The status field (ACTIVE, EXPIRED, REVOKED) is derived at runtime whenever the pass is accessed or validated, ensuring that date validity is enforced consistently without requiring constant background jobs. For QR generation, the app serialises a compact payload (at minimum, the unique pass ID and a simple integrity marker) and renders it as a QR image using a local generator library. This QR is displayed on a dedicated screen with high contrast and appropriate scaling to facilitate quick scanning on board. On the Conductor side, the camera is integrated via CameraX to provide a live preview, while Google ML Kit's Barcode Scanning API decodes incoming QR codes entirely on-device. Once the QR is decoded, the app looks up the corresponding pass in Room, checks validity dates and status, and then returns a decision in real time without a network round-trip.

To support operational transparency and future analytics, the system implements a Shift Log model tied to the Conductor role. Each successful validation triggers the creation of a ShiftLog entry capturing the conductor ID, pass ID, and timestamp, all stored locally in Room. Conductors can view summaries of their recent validations (e.g., per day or per shift) through a simple log screen, while Admins can later extend the same data model to generate route-level or conductor-level reports once backend synchronisation is introduced. Throughout development, the app was tested on mid-range Android devices with varying lighting conditions to ensure scanner reliability and acceptable performance without internet. Edge cases such as unreadable QR codes, expired passes, inactive users, and denied camera permissions were explicitly handled at the UI layer with clear error messages. This methodology results in an offline-first, modular, and extensible system that meets current demonstration needs while providing a clear path toward production-grade security and cloud integration.

V. RESULT AND DISCUSSION:

The implemented QR Code Based Bus Pass Management System successfully realises the core objectives defined in the SRS: digitising the pass lifecycle, supporting three distinct roles in a single APK, and ensuring offline operation using Room (SQLite). In controlled tests on mid-range Android devices, the application allowed Admins to create users and pass products, Passengers to register and purchase passes, and Conductors to validate QR codes without any network connectivity. Role- aware navigation via SharedPreferences functioned reliably, with users being redirected to the correct dashboard immediately after login or app relaunch. This confirmed that a single, unified APK can effectively encapsulate the workflows of Admin, Passenger, and Conductor without forcing agencies to maintain separate applications for each role.

From a performance perspective, the combination of CameraX and Google ML Kit (Barcode Scanning) yielded fast and stable QR recognition in typical indoor and outdoor lighting conditions. In practical use, pass validation—from pointing the camera at the QR code to displaying the validation result—completed within a short, user-acceptable time window, making it suitable for peak-hour operations where boarding speed is critical. Room (SQLite) queries for pass lookup and status checks executed locally and instantaneously for the tested dataset sizes, indicating that the chosen data model (users, pass_products, passes, and shift_logs) is adequate for small to medium operators who may only need to store a few thousand records per device. Importantly, the offline design avoided the latency spikes associated with network calls, which are commonly reported as pain points in online-only e-ticketing systems.

The Shift Log feature demonstrated its value in improving auditability and transparency. Each successful validation by a conductor resulted in a corresponding log entry capturing conductor ID, pass ID, and timestamp. Even in the absence of a backend server, these logs provide a tamper- resistant local trail that can be exported or synced later when a server module is added. Compared to paper-based registers and informal tallying, the digital logs offer a more consistent, machine- readable basis for reconciliation of ridership and for monitoring conductor activity. This directly addresses gaps identified in the literature, where many QR-based systems focus on ticket generation and enforcement but treat operational logging and analytics as secondary concerns.

Usability observations indicate that the Material Design 3 ‑ based interface is intuitive for all three roles after minimal orientation. Passengers were able to complete registration and pass purchase with few steps, and the dedicated ―Show QR‖ screen reduced friction during boarding. Conductors benefited from a simplified scanner interface with a prominent ―Scan‖ action and clear success/failure messages, reducing cognitive load in crowded buses. Admins could manage users and pass products from the same app without needing desktop tools. Compared to traditional paper passes and manual inspection, the prototype demonstrated qualitative improvements in boarding speed, reduction in disputes over validity (as the app clearly showed pass status and expiry), and elimination of issues related to damaged or illegible paper passes.

VI. CONCLUSION

This work presented a QR Code Based Bus Pass Management System tailored to the operational realities of Indian public transport. By consolidating three roles—Admin, Passenger, and Conductor— into a single Android APK, the system replaces paper-based passes and manual inspection with a digital, QR-enabled workflow that is both practical and affordable. Built using Java, Room (SQLite), SharedPreferences, CameraX, and Google ML Kit, the application supports end-to-end pass lifecycle management: registration, pass creation, local storage, QR generation, on-device scanning, and shift-wise logging. The offline-first architecture ensures that critical functions remain available even on routes with poor or no connectivity, addressing a key limitation of many earlier online-centric solutions.

Experimental use of the prototype on mid-range Android devices showed that QR validation and database lookups execute quickly enough for busy boarding conditions, while the Shift Log mechanism creates a tamper-resistant local audit trail of validations. The use of Material Design 3 and clear role-based dashboards reduced training requirements for bus staff and made the system accessible to typical commuters. Collectively, these results demonstrate that a lightweight mobile application can deliver immediate benefits: reduced paper use, faster boarding, fewer disputes over pass validity, and improved transparency for operators.

At the same time, the project is intentionally positioned as a foundation for further hardening and integration, not as a final commercial product. Security is demonstration-grade (plain-text credentials and unsigned QR payloads), and there is no central backend for multi-device synchronisation, online payments, or fleet-wide analytics. Future work will focus on integrating password hashing and key rotation, adding cryptographic signatures to QR codes, and building a backend API for syncing users, passes, and Shift Logs across devices and depots. Additional enhancements—such as online payment gateways, real-time dashboards, and integration with city-level smart transport platforms—can then be layered on top of the current architecture. Thus, the proposed system serves both as a working prototype and as a scalable blueprint for modernising bus pass management in resource- constrained, connectivity-challenged environments.

REFERENCES

[1] S. L. Fong, D. Wui Yung Chin, R. A. Abbas, A. Jamal and F. Y. H. Ahmed, "Smart City Bus Application With QR Code: A Review," 2019 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS), Selangor, Malaysia, 2019, pp. 34-39, doi: 10.1109/I2CACIS.2019.8825047.

[2] S. L. Fong, D. Wui Yung Chin, R. A. Abbas, A. Jamal and F. Y. H. Ahmed, "Smart City Bus Application With QR Code: A Review," 2019 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS), Selangor, Malaysia, 2019, pp. 34-39, doi: 10.1109/I2CACIS.2019.8825047.

[3] Mrs.G.SWETHA, 2J.BHAVANA QR BASED STUDENT BUS PASS SYSTEM https://ijarst.in/public/uploads/paper/323311709636 935.pdf

[4] Karale1 E- Ticket Booking System for Public Transportation using QR Code link: https://ijarsct.co.in/Paper15605.pdf

[5] N., Krishnammal & C, Ramya & Shiva, Ganesh & R, Shrenidhi. (2020). Efficient Bus Pass Generation and Authentication using QR Code. International Journal of Innovative Technology and Exploring Engineering. 9. 692-694. 10.35940/ijitee.G5304.059720.

[6] Sb, Vaidianathan. (2024). An Android-based Smart Bus Pass application. 2. 126-138.

[7] Yadav, Sayali. (2024). Integration of Modernized application of smart Bus Pass.. INTERANTIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING AND MANAGEMENT. 08. 1-5. 10.55041/IJSREM35496.

[8] Gaikwad, Ajinkya. (2021). Smart Bus: A Smartphone based Framework for Public Transport Ticket System using QR Code and Implementing Delay Payment Method. International Journal for Research in Applied Science and Engineering Technology. 9. 1507-1511. 10.22214/ijraset.2021.34583.

[9] Tupare, Jaypal & Tupe, Ajay & Saudagar, Mahebub & Mali, Sujata. (2024). Smart Public Transport Ticketing System Using QR Code Online Payment Method. SSRN Electronic Journal. 10.2139/ssrn.4843352.

DOI: 10.48175/IJARSCT-30011