

Metro Ticket Booking System

Janhavi Patil¹, Tushar Patil²

Department of Computer Engineering

R. C. Patel Institute of Technology, Shirpur, Maharashtra, India

Abstract—Metro rail systems in rapidly growing urban centers face increasing commuter volumes, and conventional counter-based ticketing continues to contribute to long queues, peak-hour congestion at station entry points, and inefficient use of station staff. This paper presents the design and development of a Metro Ticket Booking System that enables commuters to book single-journey and pass-based tickets online, generate QR-code-based digital tickets for contactless gate entry, and allows administrators to manage routes, fares, and station data through a dedicated dashboard. The system was developed using a three-tier web architecture with a relational database backend, a server-side application layer handling fare calculation and ticket generation, and a responsive front-end supporting commuter and administrator roles. Core features include route- and station-based fare calculation, QR-code ticket generation and gate-scan validation, recharge-based smart-pass management, and an administrative panel for route and fare configuration. The system was evaluated through functional testing across booking, payment, and gate-validation workflows, and load testing under simulated concurrent booking sessions. Results show that all core functional test cases passed verification, QR-code gate validation completed in an average of 0.8 seconds per scan, and the system sustained stable response times up to a simulated load of 150 concurrent booking sessions. These results indicate that a dedicated online ticket-booking and QR-based validation system can meaningfully reduce station queuing time and improve the efficiency of metro commuter throughput.

Index Terms—Metro Ticketing, QR Code Validation, Online Booking, Smart Pass, Fare Calculation, Web Application, Public Transportation, Contactless Ticketing.

I. INTRODUCTION

Urban metro rail networks have become a primary mode of mass transit in many growing cities, but conventional counter-based ticket purchase and token-based gate entry continue to be a major source of commuter delay, particularly during peak-hour travel. Long queues at ticket counters and token-dispensing machines not only inconvenience commuters but also require continuous staffing of ticketing counters at every station.

Online ticket booking combined with QR-code-based contactless gate validation offers an efficient alternative, allowing commuters to purchase tickets in advance from a mobile or web interface and pass through station gates via a quick QR-code scan, reducing both purchase-time and gate-entry time relative to traditional token systems.

This paper presents the design, development, and evaluation of a Metro Ticket Booking System supporting online single-journey and pass-based ticket booking, QR-code ticket generation, and gate-scan validation, alongside an administrative panel for route, station, and fare management.

II. LITERATURE REVIEW

III. SYSTEM DESIGN

A. System Architecture

The system was designed using a three-tier architecture comprising a responsive presentation layer (commuter booking interface and administrator dashboard), an application layer (server-side fare calculation, booking, and ticket-generation logic), and a data layer (relational database) storing route, fare, booking, and pass data.

B. Core Functional Modules

Key modules include route- and station-based fare calculation, single-journey QR-ticket booking, rechargeable smart-pass management for frequent commuters, QR-code gate-scan validation, and an administrative dashboard for route, station, and fare configuration. An overview is presented in Table I.

C. Ticket Generation and Validation

Upon successful payment, the system generates a unique QR-coded ticket encoding the journey's origin, destination, and validity window. At station gates, the QR code is scanned and validated in real time against the booking database to confirm ticket validity before permitting entry.

A. Automated Fare Collection Systems

Automated Fare Collection (AFC) systems, encompassing smart-card and token-based gate validation, have been widely studied for their role in reducing manual ticketing overhead and enabling centralized fare-revenue tracking across metro networks.

B. QR-Code-Based Ticketing

QR-code-based ticketing has emerged as a low-infrastructure-cost alternative to smart-card systems, requiring only a QR-code scanner at gates rather than dedicated smart-card readers, while still supporting fast, contactless validation suitable for high-throughput commuter gates.

TABLE I — Core Functional Modules of the Metro Ticket Booking System

Module	Primary User	Key Functionality
Fare Calculation	System (automated)	Compute fare based on origin-destination station pair
Ticket Booking	Commuter	Book single-journey ticket with online payment
Smart Pass Management	Commuter	Recharge and manage rechargeable travel pass
QR Gate Validation	System (automated)	Scan and validate QR ticket at station gate
Route/Fare Administration	Administrator	Configure routes, stations, and fare tables

IV. IMPLEMENTATION AND EVALUATION

The system was implemented as a web application using a standard three-tier stack, with the application layer handling fare computation based on configurable per-station distance-fare tables, QR-code generation upon payment confirmation, and real-time gate-validation lookups against the central booking database.

Functional testing covered booking, payment, QR-generation, and gate-validation workflows, while load testing simulated concurrent booking sessions ranging from 50 to 150 users to evaluate system responsiveness under peak-hour-representative demand.

V. RESULTS AND DISCUSSION

All core functional test cases passed verification with no critical defects identified, as summarized in Table II. QR-code gate-scan validation completed in an average of 0.8 seconds per scan, supporting high-throughput commuter flow at station gates. Under simulated load testing, the system maintained an average booking response time below 1.5 seconds up to 150 concurrent sessions, with results summarized in Table III, indicating stable performance under peak-hour-representative concurrent demand.

VI. CONCLUSION

This paper presented the design and evaluation of a Metro Ticket Booking System supporting online booking, QR-code ticketing, and gate-scan validation. Results demonstrated reliable functional behavior, fast gate-validation performance, and stable responsiveness under simulated peak load, supporting the system's practical value in reducing station queuing time and improving commuter throughput for metro rail networks.

TABLE II — Functional Test Coverage Summary

Test Case Category	Test Cases Executed	Passed	Failed
Ticket booking and payment	12	12	0
QR-code generation	6	6	0

Test Case Category	Test Cases Executed	Passed	Failed
Gate-scan validation	8	8	0
Smart pass recharge/management	7	7	0
Administrative route/fare configuration	9	9	0

TABLE III — Load Testing Results Across Concurrent Booking Sessions

Concurrent Sessions	Avg. Booking Response Time (s)	Transaction Success Rate (%)
50	0.6	99.7
100	1.0	99.3
150	1.5	98.6

VII. REFERENCES

- [1] Pressman, R.S., and Maxim, B.R. (2019). Software Engineering: A Practitioner's Approach (9th ed.). McGraw-Hill Education.
- [2] Sommerville, I. (2015). Software Engineering (10th ed.). Pearson Education.
- [3] Elmasri, R., and Navathe, S.B. (2015). Fundamentals of Database Systems (7th ed.). Pearson Education.
- [4] Kroenke, D.M., and Auer, D.J. (2015). Database Processing: Fundamentals, Design, and Implementation (14th ed.). Pearson Education.
- [5] Chandra, S., and Kumar, U. (2018). Automated fare collection systems for urban rail transit: A review. Journal of Public Transportation, 21(1), 45-58.
- [6] Denso Wave Incorporated. (1999). QR Code Essentials. Technical Documentation.
- [7] Fowler, M. (2002). Patterns of Enterprise Application Architecture. Addison-Wesley.
- [8] Menasce, D.A. (2002). Load testing of Web sites. IEEE Internet Computing, 6(4), 70-74.