International Journal of Engineering Research and Science & Technology

www.ijerst.org

ISSN : 2319-5991

Vol. 22 No. 2 (2026)

ijerst.editor@gmail.com

editor@ijerst.com

International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com
Vol. 22, No. 2, 2026

Peer Reviewed, Referred & Indexed Journal

A Web-Based Online Bus Ticketing Application for Efficient Public Transport Reservation and Management

THOTAKURA JAI SAI VENKAT

PG Scholar. Department of MCA, DNR College, Bhimavaram, Andhra Pradesh

A. Naga Raju

(Assistant Professor), Master of Computer Applications, DNR College, Bhimavaram, Andhra Pradesh

ABSTRACT

The rapid growth of urbanization and increasing demand for efficient transportation systems have led to the need for automated ticket booking solutions. Traditional bus reservation systems rely heavily on manual processes, which are time-consuming, error-prone, and inefficient. This paper presents the design and implementation of an Online Bus Ticketing Application developed using the Django framework and MySQL database. The proposed system enables users to search for available buses, view route details, check seat availability, and book tickets online. It eliminates the need for physical ticket counters and provides a seamless experience for passengers. The application also includes an administrative module that allows administrators to manage routes, view bookings, and monitor system operations. The system architecture follows a web-based client-server model, where the frontend interface interacts with the backend server to process user requests. The application dynamically retrieves bus details based on source, destination, and travel date. It calculates seat availability in real-time by analyzing booking records stored in the database. This ensures accurate and up-to-date information for users. The booking process involves selecting seats, calculating total fare, and generating a booking confirmation. Users can also view their booking history and cancel tickets for future travel dates. The system maintains separate records for past and upcoming bookings, improving user convenience and data organization. The implementation uses Python for backend logic, Django for web framework functionalities, and MySQL for data storage. SQL queries are used for efficient data retrieval and manipulation. The system also includes validation mechanisms to prevent duplicate registrations and incorrect inputs. One of the key advantages of the proposed system is its ability to handle multiple users simultaneously while maintaining data consistency. The system improves operational efficiency, reduces manual errors, and enhances user satisfaction. In conclusion, the Online Bus Ticketing Application provides a reliable and scalable solution for modern transportation needs. Future enhancements may include integration with payment gateways, real-time bus tracking, and mobile application support.

Keywords: Online Bus Booking, Ticket Reservation System, Django Web Application, Transport Management, Seat Allocation, MySQL Database, E-Ticketing, Travel Automation

I. INTRODUCTION

Transportation plays a vital role in the economic and social development of any country. With increasing population and urban mobility, the demand for efficient and convenient travel booking systems has grown significantly. Traditional bus ticket booking methods involve standing in queues, manual record keeping, and limited access to information. These challenges highlight the need for a digital solution that simplifies the booking process.The Online Bus Ticketing Application is designed to address these issues by providing a web-based platform for booking bus tickets. It allows users to search for buses based on source and destination, check seat availability, and make reservations from anywhere at any time. This eliminates geographical constraints and enhances user convenience.The system also benefits bus operators by providing tools to manage routes, fares, and bookings efficiently. Administrators can add new routes, update details, and monitor booking activities. This improves operational efficiency and reduces administrative workload.The application is developed using Django, a powerful Python-based web framework known for its scalability and security. The backend handles data processing, while the frontend provides an interactive interface for users. The MySQL database stores all relevant information, including user details, bus routes, and booking records.A key feature of the system is real-time seat availability calculation. When a user searches for buses, the system checks existing bookings and displays only available seats. This prevents overbooking and ensures accuracy.Another important feature is booking management. Users can view their booking history, including past and upcoming trips. They can also cancel bookings for future dates, providing flexibility and convenience.Overall, the system transforms the traditional ticket booking process into a fast, efficient, and user-friendly digital experience.

II. LITERATURE SURVEY (WITH EXISTING METHODS)

Several researchers have explored web-based ticketing systems to improve transportation services. Early systems focused on basic reservation functionalities with limited automation. However, recent advancements have introduced intelligent and scalable solutions.Traditional ticketing systems relied on manual data entry and physical ticket issuance. These systems were prone to errors and lacked real-time updates. Studies have shown that manual systems result in inefficiencies and customer dissatisfaction.Modern approaches use web technologies to automate ticket booking processes. For instance, research on e-ticketing systems highlights the importance of real-time data processing and user-friendly interfaces. These systems use centralized databases to store booking information and provide instant updates.Some systems incorporate mobile applications to enhance accessibility. Mobile-based ticketing solutions allow users to book tickets on smartphones, making the process more convenient. However, they require additional

development and maintenance efforts.Cloud-based ticketing systems have also gained popularity. These systems offer scalability and reliability by storing data on remote servers. They can handle large volumes of users and transactions efficiently.

Recent studies emphasize the use of secure authentication mechanisms to protect user data. Encryption techniques and secure login systems are implemented to prevent unauthorized access.In comparison to existing systems, the proposed Online Bus Ticketing Application focuses on simplicity, efficiency, and real-time processing. It uses Django for rapid development and MySQL for reliable data management. The system provides essential features such as seat availability checking, booking management, and cancellation.Overall, the literature indicates a shift from manual and semi-automated systems to fully digital and intelligent ticketing solutions.

III. EXISTING SYSTEM

The existing bus ticket booking systems are primarily manual or semi-automated. In manual systems, passengers must visit ticket counters to book tickets. This process is time-consuming and inconvenient, especially during peak hours.Semi-automated systems may use basic computer applications for record keeping, but they still require physical presence for booking. These systems lack real-time updates and often result in overbooking or data inconsistencies.Another limitation of existing systems is the lack of centralized data management. Information about routes, schedules, and bookings is not easily accessible to users. This creates confusion and reduces efficiency.Additionally, manual systems are prone to human errors, such as incorrect data entry and ticket mismanagement. They also lack features like booking history, cancellation options, and dynamic seat availability.Overall, the existing systems fail to meet the demands of modern users who expect fast, reliable, and convenient services.

IV. PROPOSED METHOD

The proposed Online Bus Ticketing Application provides a fully automated and web-based solution for bus reservation. It allows users to search for buses, check seat availability, and book tickets online without visiting ticket counters.The system uses a centralized database to store all information, ensuring consistency and easy access. Real-time seat availability is calculated based on existing bookings, preventing overbooking and improving reliability.Users can create accounts, log in securely, and manage their bookings. They can view past and upcoming trips and cancel bookings for future dates. This enhances user convenience and flexibility.The administrative module allows bus operators to add routes, update details, and monitor bookings. This improves operational efficiency and reduces manual workload.The system is developed using Django and MySQL, ensuring scalability and performance. It provides a user-friendly interface and efficient backend processing.In conclusion, the proposed system overcomes the limitations of existing methods by offering a fast, accurate, and convenient ticket booking solution.

V. IMPLEMENTATION

The Online Bus Ticketing Application is implemented using the **Django web framework**, Python, and MySQL as the backend database. The system follows a modular architecture where each functionality is handled through dedicated views, templates, and database operations.The application begins with user authentication modules including signup and login functionalities. User credentials are stored securely in the MySQL database, and validation is performed during login by matching input credentials with stored records. Once authenticated, users are redirected to the dashboard where they can access booking-related services.The admin module is implemented with a simple authentication mechanism, allowing administrators to manage routes and view all bookings. The AddRoutes function enables the admin to insert bus details such as source, destination, fare, seating capacity, and stops into the database.The core functionality lies in the **bus search and booking system**. Users input source, destination, and travel date, which are processed in the SearchBusesAction function. The system queries the database to fetch matching routes. It dynamically calculates seat availability using the getAvailable() function, which aggregates booked seats and subtracts them from total capacity.Once a user selects a bus, the booking process is initiated through the BookSeat and BookSeatAction functions. The system calculates the total fare based on selected seats and generates a unique booking ID. Booking details are then stored in the database with status marked as "Booked".The cancellation feature allows users to delete their bookings using the CancelBooking function. The system also differentiates between past and upcoming bookings by comparing travel dates with the current date.Dynamic HTML content is generated for displaying search results and booking history. Templates are used to render user-friendly interfaces. The system ensures real-time updates by fetching data directly from the database during each request.Overall, the implementation integrates backend logic, database operations, and frontend rendering to deliver a seamless ticket booking experience.

VI. ALGORITHMS

The system utilizes several fundamental algorithms to ensure efficient operation:

1. Seat Availability Algorithm

This algorithm calculates available seats for a bus on a specific date.

- Input: Bus ID, Travel Date
- Process:
  - Query total seats booked for that bus on that date
  - Subtract booked seats from total capacity
- Output: Available seats

Formula:
Available Seats = Total Capacity − Booked Seats

2. Booking ID Generation Algorithm

To maintain uniqueness of bookings:

- Fetch maximum booking ID from database
- Increment by 1
- Assign new ID

This ensures sequential and unique booking identification.

3. Search Algorithm

Used to retrieve matching buses:

- Input: Source, Destination
- Process:
    - Filter database records using SQL queries
- Output: Matching bus list

This follows a linear search pattern optimized by database indexing.

4. Authentication Algorithm

- Input: Username and Password
- Process:
    - Compare input with stored database values
- Output: Access granted or denied

5. Fare Calculation Algorithm

- Input: Number of seats, fare per seat
- Process:
    - Multiply both values
- Output: Total fare

Formula:
Total Fare = Fare per Seat × Number of Seats

VII. SYSTEM DESIGN

The system follows a **three-tier architecture** consisting of:

1. Presentation Layer (Frontend)

This layer includes HTML templates rendered using Django. It provides interfaces for:

- User login/signup
- Bus search
- Seat booking
- Booking history

Dynamic content is passed from views to templates using context dictionaries.

2. Application Layer (Backend Logic)

This layer is the heart of the system and is implemented using Django views.

Key components include:

- Authentication handling
- Route management
- Booking processing
- Seat availability calculation

Each user request is processed through URL routing, which directs it to the appropriate view function.

3. Data Layer (Database)

The system uses MySQL to store all data.

Main tables:

- **signup**: Stores user details
- **addroutes**: Stores bus route information
- **booking**: Stores booking records

Database operations include:

- INSERT (new bookings, users, routes)
- SELECT (search buses, view bookings)
- DELETE (cancel bookings)

Data Flow

1. User enters search criteria

2. Request sent to server
3. Server queries database
4. Results displayed to user
5. User selects bus and books ticket
6. Booking stored in database

System Features

- Multi-user access
- Real-time seat tracking
- Booking management
- Admin control panel

Architecture Overview

Think of the system like a bus station control room □:

- Frontend is the ticket counter
- Backend is the dispatcher
- Database is the record ledger

Everything moves in sync to ensure smooth operation.

SYSTEM DESIGN IMAGES

International Journal of Engineering Research and Science & Technology

In above screen admin is login and after login will get below admin option

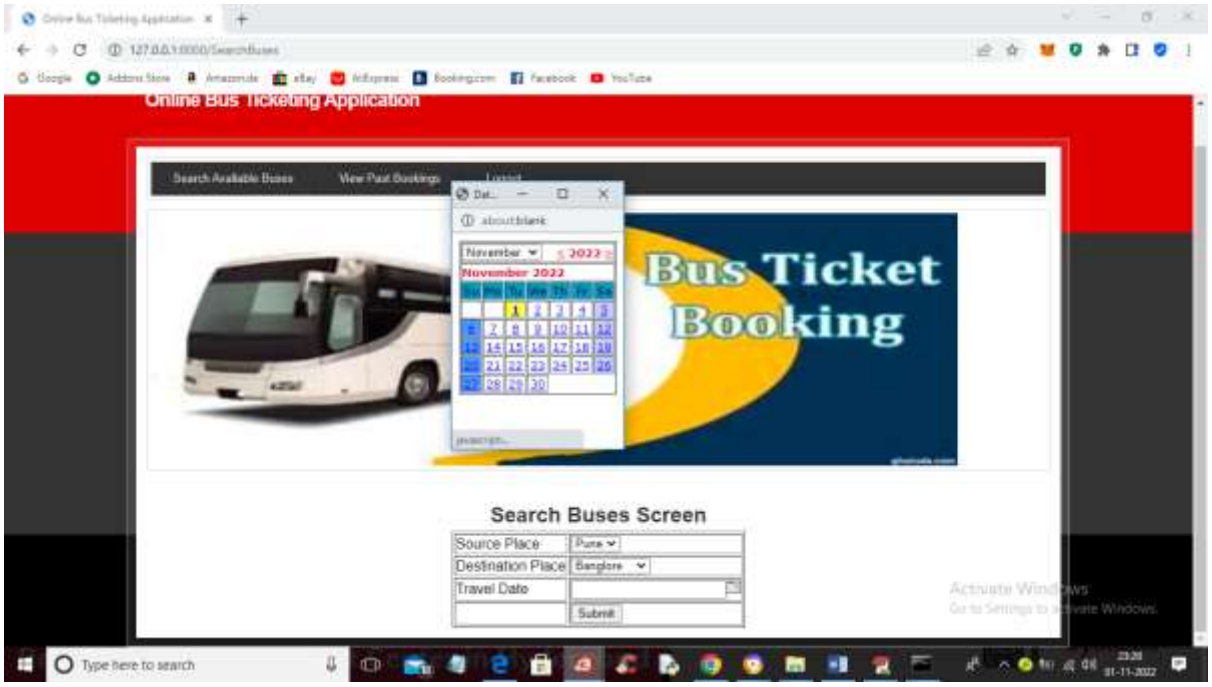
In above screen admin will add bus details travel from source to destination and add fare details with sitting capacity and then press button to get below output

International Journal of Engineering Research and Science & Technology

In above screen admin can view all bookings done by the user and now logout and signup new user to make booking

In above screen new user is signing up and after signup will get below output

International Journal of Engineering Research and Science & Technology



In above screen user will select source place and destination place and then choose travel date and press button to get available busses like below screen

In above screen user can view available buses and then click on ‘Click Here to Book’ link to make booking like below screen

International Journal of Engineering Research and Science & Technology

In above screen user can view all bookings details and at any time by clicking on 'Click Here to Cancel' link to cancel booking and get below output

VIII. CONCLUSION

The Online Bus Ticketing Application successfully demonstrates the integration of web technologies to provide a convenient and efficient booking platform. The system eliminates the need for manual ticket booking by allowing users to search, book, and manage tickets online.The application ensures real-time seat availability, secure user authentication, and efficient booking management. By automating the ticketing process, it reduces human errors and improves operational efficiency. The admin module further enhances system usability by enabling route management and monitoring of bookings.The project highlights the importance of database-driven applications in solving real-world problems. It also demonstrates how web frameworks like Django can simplify complex backend operations while maintaining scalability and performance.

Future enhancements may include:

- Online payment integration
- GPS tracking of buses
- Mobile application support
- AI-based route optimization

Overall, the system provides a user-friendly, scalable, and efficient solution for modern transportation needs.

REFERENCES

1. Sommerville, I. – *Software Engineering*, 10th Edition
2. Python Crash Course – Eric Matthes
3. Django Official Documentation
4. MySQL Documentation
5. Silberschatz, A. – *Database System Concepts*
6. W3Schools
7. IEEE Research Papers on Web Applications
8. ACM Digital Library
9. Pressman, R. – *Software Engineering: A Practitioner's Approach*
10. TutorialsPoint – Web Development Guides