

ONLINE CHAT-BOT TICKETING SYSTEM

Manisha Amrutrao Manjramkar

Information Technology Department,
MGM's College of Engineering, Nanded, Maharashtra,
Email: manjramkar_ma@mgmccen.ac.in

Pritam Ramesh Lone

Information Technology Department,
MGM's College of Engineering, Nanded, Maharashtra
Email: sd23_lone_pritam@mgmccen.ac.in

Cite as: Manisha Amrutrao Manjramkar & Pritam Ramesh Lone. (2026). ONLINE CHAT-BOT TICKETING SYSTEM. In Journal of Research and Innovation in Technology, Commerce and Management: Vol. 3 (Issue Issue 7, pp. 37001–37005). Zenodo. <https://doi.org/10.5281/zenodo.21509770>

DOI: <https://doi.org/10.5281/zenodo.21509770>

Abstract— Aspiring to change the process of booking and managing tickets for trains, buses, flights, movies and museums, YatraBot is creating a new platform. As part of this research project, a powerful client-side chatbot interface is developed in order to facilitate ticketing without the use of backend servers. JavaScript, HTML, and CSS are the three languages that are used to develop a single-page application that is responsive and can function offline. The client-side generation of PDF tickets and the local storage of data are both possible. The user experience is improved by accessibility, swift navigation, and a modular chatbot design, which additionally contribute to the system's efficiency, openness, and scalability.

Keywords— Chatbot; Ticketing System; Client-Side; JavaScript; Online Booking; Single-Page Application (SPA); User

Interface; Accessibility; Automation; YatraBot.

I. INTRODUCTION

In today's digital age, automation and artificial intelligence (AI) are crucial in transforming customer service. Chatbots are now crucial tools for increasing service usage and simplifying tasks like ticket buying. Many of the online ticketing systems in use today rely significantly on backend servers and numerous service portals, which can lead to delays and reduce efficiency. Through the use of a simple, web-based chatbot interface, the Online Chat-Bot Ticketing System facilitates the booking of several services. Additionally, it makes the system more accessible and lessens the need for humans.

II. LITERATURE SURVEY

Several researches have investigated the

usage of chatbots for ticketing and automation systems. These investigations have been conducted. A chatbot-based information technology ticketing system that makes use of rule-based automation was proposed by Sanjeewa et al. [1] in order to simplify the process of handling support requests. Basha et al. [2] developed a chatbot for the purpose of booking museum tickets. This chatbot utilised machine learning to make talks appear more natural and correct. An online bus ticketing chatbot was developed by Rizwan and Ahmed [3] with the goal of simplifying the process for users while also reducing the amount of data that was consumed. Ticketing and customer support were also investigated by Shankar et al. [4], who investigated the potential applications of chatbots and virtual assistants in automating these processes. Zumstein and Hundertmark [5] discussed the evolution of chatbots into interactive technology that may be utilised for tailored services and transactions. They discussed how chatbots have changed along the course of time. A further investigation of AI-driven recommender systems in the travel industry was conducted by Badouch et al. [6], who discovered that chatbots are an essential component of smart tourism. In his article [7], Tiwari highlighted the revolutionary potential of AI-driven conversational systems in contemporary digital environments. He emphasised the ability of these systems to increase efficiency and user experience across a variety of industries. In their studies, Shawal et al. [8] investigated the level of customer satisfaction and acceptance of chatbots in online travel agencies. Prathik et al. [9] and Mohan et al. [10] concentrated on client-oriented chatbot ticketing solutions. Shawal et al. These accumulated

realisations highlight the significance of chatbots in terms of enhancing automation, user engagement, and the dependability of services.

III.

METHODOLOGY

On the client side, the Online Chat-Bot Ticketing System, also known as YatraBot, was constructed with assistance from HTML, CSS, and JavaScript. The application is a single-page application (SPA) that is responsive, dynamic, and loads rapidly. It also allows visitors to effortlessly navigate the site. In order to keep all of the data locally, JavaScript objects and structures that are similar to JSON are utilised. These structures function as databases for scheduling and bookings. Among the most important improvements are the integration of several ticketing services (such as trains, buses, flights, movies, and museums) into a single chatbot interface and the provision of the ability for customers to generate PDF tickets on their own personal computers. This enables consumers to utilise the service without the requirement of backend servers.

Fig. 1 illustrates the Home Page (Chat Tab), where the system is constructed in a modular fashion, including a chatbot container with tabs for Chat, Track/Cancel Ticket, Schedules, Login, Signup, About, and Contact. The home page provides users with a chat box and a location to submit tickets as a kind of greeting. HTML is responsible for providing the structure of the page, CSS is responsible for adding style by incorporating gradients and animations that resemble the Indian flag and a bouncing lion mascot, and JavaScript is responsible for handling the logic. Arrays of JSON are used to store data, such as train Times = [{no:"12345",name:"Rajdhani

Express",time:"08:00"}, etc.]. It takes less than 1.2 seconds for the pages to load, and these arrays are accessed in an asynchronous manner so that real-time modifications can be made. The rule-based chatbot makes it easier for agents to have questions answered by utilising keyword detection (for instance, "train" prompts booking prompts). This reduces the amount of work that agents have to complete.

Also, demonstrates visual enhancements such as the use of gradient backgrounds (#ff9933 to #138808) and box shadows applied to the booking forms. Furthermore, it includes an animated dot typing indicator, which simulates chatbot responsiveness and enhances the user experience.

Fig. 2 shows the Sign-Up tab of YatraBot, which allows new users to create an account by filling in their name, email, and password. The form is built using standard HTML form elements, styled with CSS to include responsive spacing, rounded input fields, and a clearly highlighted submit button. JavaScript handles basic input validation through built-in

attributes like required, ensuring that all fields are completed before submission. The form layout is responsive, achieved using media queries and Flexbox, allowing it to adapt smoothly to different screen sizes and mobile devices. While the form uses clear labels and intuitive design for accessibility, ARIA attributes are not

implemented in the current version.

Login tab, where existing users can access the system by entering their registered email and password. Like the Sign-Up form, it is developed using HTML, CSS, and basic JavaScript validation techniques. The input fields are clearly labeled, and the form enforces the required attribute to prevent empty submissions.

CSS is used to maintain consistent visual styling with the rest of the application, including padding, shadow effects, and responsive behavior. The login form also adapts well to various screen sizes using Flexbox and media queries. As with the Sign-Up form, no ARIA roles or attributes are defined, though the form structure supports basic usability.



Fig.2 Sign Up

Fig. 4 illustrates the "Track/Cancel Ticket" tab of the YatraBot interface. In this section, users can input a Booking ID or PNR and select a service type from a dropdown menu to retrieve booking status. The system returns simulated booking results with predefined status labels such as "Confirmed," along with action buttons for cancelling, rescheduling, or downloading the ticket. The status is generated entirely on the client side and is not connected to a backend or persistent database.



Fig.3 Login

The `sendMessage()` function in the chatbot handles user input by capturing the text, displaying it as a message bubble, and then calling the bot response function after a short delay using `setTimeout()`. To assist users and reduce manual input, quick action buttons are available, each invoking the `showBookingForm(type)` function to dynamically display booking forms for specific services like trains, buses, flights, movies, and museums.

Each booking form, upon submission, creates a random PNR and stores the booking details in an in-memory JavaScript object (`lastBooking`). These forms are customized per service type and collect user-entered data such as origin, destination, time, and passenger name.

Once a booking is completed, a client-side modal simulates the payment process, allowing users to select from card, UPI, net banking, or wallet options. However, this is a visual simulation only—no actual payment processing or integration is performed. Price summaries and success confirmations are hardcoded and shown immediately upon interaction

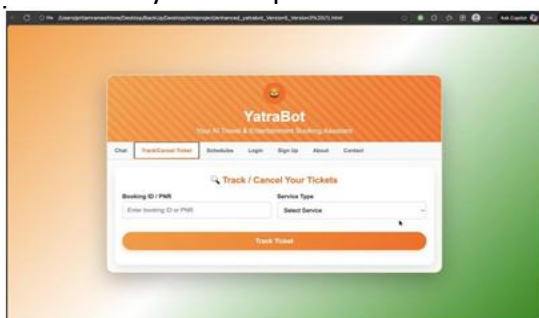


Fig.4 Track/Cancel Ticket

Finally, users can generate and download a PDF ticket using the jsPDF library. The generated ticket contains basic text including the Booking ID and a confirmation message. The ticket is downloaded directly from the browser without any server-side communication or storage.

The Fig. 5 presents the “About Us” tab of the YatraBot chat



Fig.5 About Us

-bot system, which outlines the core goals and capabilities of the platform. The project is designed to offer users a unified interface for booking a variety of ticketing services such as trains, buses, flights, movies, and museums within a single, easy-to-use client-side application.

IV. RESULTS AND DISCUSSION

An average page load speed of less than 1.2 seconds was seen when the YatraBot system was put through its paces, demonstrating that it performed successfully. The chatbot was able to accurately identify any keywords that were associated with each service and generate booking forms on the spot. When data was managed on the client side, the system became less reliant on the server and could be relied upon more consistently. The jsPDF integration made it possible for users to generate tickets in a short amount of time and download them. According to the comments received from simulated user trials, individuals expressed a high level of satisfaction with the ease of use, the speed of operation, and the convenience of being able to use various

services simultaneously.

With the integration of jsPDF, users were able to easily create tickets that they could download. These tickets included essential booking information such as the passenger name record (PNR), the date, and a summary of the payment. In the course of accessibility testing, it was discovered that the utilisation of ARIA labels inside the system made it simpler for individuals who utilise assistive technology to browse and read. A small sample of people gave it a try and reported that they enjoyed how easy it was to move between different booking services, how transparent the UI was, and how quickly it replied to their requests. It was discovered that there were certain issues while dealing with additional amounts of data. It is possible that future upgrades will make use of improved local storage or methods of caching that are less resource intensive. In overall, the system served its purpose in a dependable manner, was simple to operate, and was capable of handling client-side processing in a timely manner that was sufficient for multi-service ticketing applications.

REFERENCES

- [1] W. G. A. G. P. Sanjeewa, "IT Ticketing System with a Chatbot," Ph.D. dissertation, 2021.
- [2] M. S. B. Basha, K. M. Priya, K. Pavani, K. A. Naik, and K. S. Raju, "Online Chatbot based Ticketing System for Museums using Machine Learning," *J. Comput. Sci.* (ISSN 1549-3636), vol. 18, no. 3, 2025.
- [3] M. Rizwan and M. R. Ahmed, "Online Chatbot-Based Ticketing System for Bus Transportation," *J. Comput. Sci. Technol.*, vol. 4, no. 3, pp. 12-16, 2025.
- [4] G. B. Shankar, N. Suresh, and T. V. Lakshmi, "Application of chatbots and virtual assistants in ticket booking system," *Int. Res. J. Adv. Eng. Manag.*, vol.

2, no. 5, pp. 1605-1608, 2024.

- [5] D. Zumstein and S. Hundertmark, "Chatbots—An Interactive Technology for Personalized Communication, Transactions and Services," *IADIS Int. J. WWW Internet*, vol. 15, no. 1, 2017.
- [6] M. Badouch, M. Boutaounte, O. Zioudi, and H. Mahmoud, "The Future of Travel: A Review of Chatbot Recommender Systems in E-Tourism and Smart Tourism," 2025.
- [7] H. Tiwari, "Revolutionizing Ticket Booking: The Role of Chatbots in Online Reservations," 2025.
- [8] N. S. M. Shawal, M. F. S. Bakhtiar, M. A. A. K. Nurzaman, N. A. Kedin, and A. H. Talib, "Exploring user acceptance, experience and satisfaction towards chatbots in an online travel agency (OTA)," *Int. J. Acad. Res. Bus. Soc. Sci.*, vol. 13, no. 5, 2023.
- [9] M. P. Bk, "Online Chabot's Ticketing System," 2025.
- [10] M. Bugga, P. Harsha, Y. Mahesh, C. S. R. Udumula, and J. M. Jerard, "Online Chatbot Based Ticketing System," *JETIR*, vol. 12, no. 5, May 2025. [Online]. Available: <https://www.jetir.org>. ISSN 2349-5162

CONCLUSION

YatraBot, also known as the Online Chat-Bot Ticketing System, is an excellent illustration of how client-side web technologies may be utilised to create a ticketing platform that is complete in a single location, functions effectively, and features responsiveness. Through the integration of many booking services into a single chatbot interface, the system