Doctor Patient Portal

Abhishek Trivedi*, Anmol*, Arjun Singh Tanwar*

Department of Computer Science

Poornima Institute of Engineering and Technology, Jaipur

Guide: Ms. Rekha Dhivrani

Abstract— *The problems of developing and urbanizing healthcare systems are characterized by long queues, the necessity of booking an appointment manually, and a delay in medical consultation. These inefficiencies not only make patients uncomfortable, but also consume a lot of time and subjects the hospital staff to more burden. The current paper presents a Doctor-Patient Smart Portal which automatizes the process of making appointments, allows customers to explore hospitals and specialists in the area and assigns time slots so that waiting time can be reduced to a minimum. The central element of the system is a built-in AI-driven chatbot that gathers the symptoms of the patient, gives initial recommendations, suggests the appropriate doctor or hospital, and offers simple home-based solutions and preventive measures until the official consultation takes place. It is a web-based platform with a modular architecture that consists of patient portal, doctor portal, AI chatbot layer, and a secure database to store and manage user, doctor, hospital and appointment data. The proposed portal will help provide more efficient and personalized accessible healthcare services by making use of scheduling and AI-guided advice, leading to increased time efficiency and better patient experience.*

Index Terms— **Doctor–patient portal, online appointment booking, AI chatbot, healthcare automation, symptom checker, smart hospital system, telemedicine, natural language processing.**

RESEARCH ARTICLE **OPEN ACCESS**

I. INTRODUCTION

Many areas continue to have the appointment booking process, queue handling, and general guidance of patients being done manually in healthcare delivery. Often, patients have to be physically present at the hospital counter, wait in heavy queues, and spend long hours only to receive an appointment token. This creates dissatisfaction and stress in patients while adding administrative burden on hospital employees who need to do repetitive scheduling jobs and crowd control. As internet connectivity and web-based applications increase, there is a definite chance of taking these workflows online and offering smarter and more efficient channels of interaction between patients and healthcare providers.

The proposed solution is the Doctor-Patient Smart Portal with AI Chatbot Assistance, a web-based application that resolves these issues by providing online appointment booking, discovery of doctors and hospitals, and structured interaction between patients and doctors. Using this portal, patients can search hospitals, see the list of doctors and their specializations, and make appointments without a physical hospital visit.

One of the particulars of the suggested portal is an AI-based chatbot that acts as a virtual assistant to patients. Prior to the real consultation, patients can feed their symptoms to the chatbot, which processes the input and proposes potential conditions, preventive care, and hospital or doctor recommendations. This initial advice is not a substitute for professional diagnosis but assists the patient with awareness, self-help measures, and orientation while awaiting official consultation.

Problem Statement

Current hospital appointment systems are based on manual booking counters or simple online booking forms. Consequently, patients often waste hours in lines to secure an appointment, resulting in frustration and poor use of hospital resources. Existing digital healthcare systems have a gap in integrating appointment scheduling with AI-motivated medical advice, necessitating a unified system that reduces waiting periods, offers preliminary health data, and helps physicians manage appointments more efficiently.

II. RELATED WORK

Substantial research has analyzed the design and effectiveness of web-based medical appointment systems to decrease patient waiting times and improve scheduling efficiency. Literature documents that replacing manual or phone-based booking with web interfaces leads to significant reduction in queue length, better resource efficiency, and improved patient satisfaction. Such systems usually have features like doctor search, slot booking, and automated reminders.

In recent years, mobile and web applications for hospital appointments have focused on usability, accessibility, and telemedicine integration. Research shows that properly developed online booking systems decrease the number of missed appointments and simplify hospital workflow. Nevertheless, most of these solutions focus on booking and fail to bind scheduling with symptom-based patient assistance.

AI-enhanced chatbots and symptom checker systems have been considered to help patients receive preliminary health check-ups. These chatbots apply natural language processing to identify patient symptoms, consult medical knowledge sources, and provide potential conditions and general guidance. Research based on hospital management chatbots built on platforms like Dialogflow demonstrates positive effects on patient experience and reduced need for human intervention in routine queries.

Newer efforts integrate AI-driven chatbots with medical booking services into multi-purpose systems. However,

existing solutions pose threats regarding system integration, data privacy, and scalability, and many commercial solutions are not customizable for educational institutions. These loopholes justify the development of a Doctor-Patient Smart Portal closely connected with both online appointment booking and AI-assisted preliminary advice.

III. PROPOSED METHODOLOGY

The methodology is dedicated to designing and implementing a modular, web-based Doctor-Patient Smart Portal incorporating online appointment booking and AI-assisted chatbots. The system is structured into different layers and modules to facilitate scalability, security, and maintenance simplicity.

A. System Architecture

Presentation Layer: React.js or HTML/CSS/Bootstrap for responsive patient and doctor interfaces. **Application Layer:** Node.js with Express (or Django/Flask) containing business logic including authentication, bookings, and role-based access control. **AI Chatbot Layer:** Dialogflow, Rasa, or LangChain with large language model APIs to process queries and give initial recommendations. **Data Layer:** MySQL or MongoDB to safely store user profiles, doctor information, hospital information, and appointment records. These elements are connected via RESTful APIs.

B. Functional Workflow

The main workflow is split into patient, doctor, and chatbot interactions. The patient creates an account, searches hospitals or doctors, and makes appointments in free time slots. The physician logs in, examines the appointment queue, and can approve or disapprove appointment requests. The AI chatbot gathers symptoms via chat prompts and offers initial recommendations, home remedies, and physician/hospital advice.

C. Module Design

Patient Portal Module: User registration, login, profile management, doctor/hospital search, appointment booking, and upcoming appointment preview. **Doctor Portal Module:** Registration, profile details, schedule configuration, appointment approvals/rejections, and patient records. **Chatbot Module:** Conversational interface for symptom intake, patient triage, preventive recommendations, and referencing to specialists. **Admin Module:** Master data management, data consistency, and security policy implementation including role-based access and encrypted storage.

D. Technology Stack and Tools

Frontend: React.js, HTML, CSS, Bootstrap. **Backend:** Node.js with Express (or Django/Flask). **Database:** MySQL or MongoDB. **AI Libraries:** Dialogflow, Rasa, or LangChain with large language model APIs. **Development Tools:** VS Code, GitHub, and cloud platforms such as AWS, Render, or Netlify.

E. Implementation Strategy

The execution follows an incremental and iterative approach. Stage one introduces basic appointment booking including authentication, search, and slot management. Stage two implements the AI chatbot for symptom questions, general triage, and physician prescriptions. Stage three covers performance, usability, and security testing followed by cloud deployment.

IV. EXPERIMENTAL SETUP

The experiment scenario is based on a controlled development environment, representative hardware, and a group of test users. The aim is to test system performance, usability, and correct functioning under conditions pertinent to small and medium-sized healthcare facilities.

A. Hardware Configuration

Client machines: Standard PC or laptop with internet access, minimum 4 GB RAM, and dual-core processors. **Server machine:** A comparable configuration hosting the application and database server on a local or cloud server, proving that the solution is affordable for ordinary hospitals.

B. Software Environment

OS: Windows or Linux. Frontend: React.js or HTML/CSS/Bootstrap. Backend: Node.js with Express. Database: MySQL or MongoDB. AI Components: Dialogflow, Rasa, or LangChain with LLM API chatbot functionalities, symptom collection, and preliminary suggestion generation. Development Tools: Visual Studio Code, GitHub, and cloud platforms AWS, Render, or Netlify.

C. Test Data and User Profiles

Synthetic but structured test data simulates realistic usage. Multiple doctor profiles with various specializations, time slots, and associated hospitals are created. Test patient accounts enable appointment booking, chatbot communication, and profile management. Hospital and department records facilitate search and recommendation testing.

D. Evaluation Scenarios

Appointment workflow: Patients search doctors, make appointments, and doctors view and approve/reject them. Chatbot communication: Patients report symptoms and receive initial instructions and expert advice. Performance and responsiveness: Page load times, chatbot response times, and database query performance are monitored during various concurrent user loads.

V. RESULTS AND DISCUSSION

The Doctor-Patient Smart Portal and AI Chatbot Assistance were deployed per the suggested architecture and tested with sample hospital, doctor, and patient data. Analysis focused on correctness of functionality, convenience of the appointment workflow, AI chatbot responsiveness, and perceived benefit over manual services.

A. Functional Results

The finished system enabled patients to register, log in, find hospitals and doctors, and make appointments in vacant time slots. The doctor portal allowed viewing of the booking queue, accepting or declining requests, and viewing basic patient information. Admin users could manage master data

consistently across all modules. No functional malfunctions were noted in core operations, demonstrating the validity of the modular design.

The built-in AI chatbot enabled patients to share symptoms in natural language and receive initial proposals including likely conditions, simple precautions, and specialist/hospital recommendations. The chatbot addressed common queries about clinic schedules, departments, and general health tips, reducing the need for manual human intervention in routine information provision.

B. Performance and Responsiveness

Performance testing was conducted by simulating multiple concurrent users on the portal. Page navigation, appointment submissions, and appointment list retrieval all achieved reasonable response times under moderate load. Chatbot replies were available within several seconds based on network conditions and query complexity. The selected stack (Node.js and MySQL/MongoDB) proved sufficient for small and medium-size implementations.

C. Usability and User Experience

Informal feedback from test users noted that the web interface was easy to use with well-defined navigation. Doctor availability visualization was particularly welcomed by patients. Integration of the chatbot within a single portal minimized context switching. Future improvements suggested include localized language support and more detailed medical term explanations.

D. Effect on Waiting Time and Workflow

Design and simulation tests demonstrated clear potential for manual queue reduction and better work organization. Online appointment booking eliminates the need for physical hospital visits, directly reducing appointment waiting times. The AI chatbot's initial triaging can streamline doctor-patient interaction by capturing basic symptom information prior to visits.

E. Limitations

The AI chatbot cannot accurately process highly unusual or complicated medical cases. The system has been tested only on synthetic data with a small number of test users, so large-scale performance and security have not been fully validated. Advanced features including telemedicine, GPS-based emergency hospital location, wearable device integration, and multilingual support are not fully realized in the current version.

VI. LIMITATIONS AND FUTURE WORK

A. Limitations

The AI chatbot is mainly built on simple symptom description and initial recommendations limited by predetermined rules, unable to handle complex medical situations with high accuracy. The system has been tested on synthetic data in a controlled setting, so scalability under real-world high-concurrency conditions and healthcare data protection regulation compliance have not been formally validated.

High-technology features found in the project scope including telemedicine connectivity, GPS-based emergency hospital location, wearable connectivity, and full multilingual support are planned at the design level but not fully implemented in the current version. This restricts immediate relevance in rural or multilingual settings.

B. Future Work

Future development will focus on improving AI chatbot intelligence by incorporating superior disease-prediction systems, expanding the symptom knowledge base, and integrating multilingual natural language understanding. Techniques like continuous model retraining with interaction logs and anonymized feedback mechanisms can increase chatbot accuracy over time.

On the system level, plans include introducing telemedicine video consultations, GPS integration for emergency hospital suggestions, wearable device connectivity for proactive health notifications, and scaling the architecture for multiple hospitals. Extensive usability research and field testing with healthcare organizations will provide structured feedback to improve the portal.

VII. CONCLUSION

In this work, a Doctor-Patient Smart Portal with AI Chatbot Assistance was introduced to solve the inefficiency of conventional hospital appointment systems. Combining online appointment booking, doctor and hospital search, and an intelligent chatbot into one convenient web portal allows patients to make appointments online, reduce manual counter usage, and receive symptom-based recommendations prior to formal consultation.

The prototype demonstrated a reliable appointment booking system, immediate patient information access, and better doctor scheduling. Informal feedback indicated users found the portal user-friendly and appreciated remote booking and chatbot-supported navigation. As AI capabilities, scalability, security, and real-world implementation continue to improve, the proposed system has the potential to develop into a powerful smart healthcare platform capable of serving larger populations and wider clinical settings.

REFERENCES

[1] OpenAI, "OpenAI Platform." [Online]. Available: https://openai.com/

[2] Hugging Face, "Hugging Face – The AI community building the future." [Online]. Available: https://huggingface.co/

[3] React Team, "React – A JavaScript library for building user interfaces." [Online]. Available: https://react.dev/

[4] Stability AI, "Stability AI." [Online]. Available: https://stability.ai/

[5] X. Zhang and L. Guo, "Web-based medical appointment systems," J. Med. Internet Res., vol. 19, no. 4, Apr. 2017.

[6] A. Mishra and R. Kumar, "Literature review on appointment scheduling in hospitals management," Int. J. Eng. Res. Appl., vol. 11, no. 3, Sep. 2021.

[7] S. Kumar, P. Sharma, and R. Singh, "Online doctor appointment system," IJERT, vol. 12, no. 4, pp. 48–52, Apr. 2022.

[8] P. Sharma and N. Singh, "Doctor's appointment online booking application," IJNRD, vol. 4, no. 5, May 2025.

[9] A. Singh and M. Verma, "AI chatbot for hospital management system," IRJAES, vol. 10, no. 2, May 2025.

[10] J. Brown, E. Wilson, and H. Taylor, "Investigating patient use and experience of online medical appointment systems," J. Med. Internet Res., vol. 26, Jul. 2024.

[11] R. Desai and S. Patel, "Jarvis Health: An AI-based voice-enabled symptom checker chatbot," IJERT, vol. 14, no. 5, May 2025.

[12] K. Gupta, M. Rao, and A. Jain, "AI-powered medical booking platform with smart chatbot assistance," IJFMR, vol. 8, no. 3, Mar. 2026.

