HOSPITRACK – A Cloud Based Hospital Management System

Mr. Vaduganathan D[1] , Arunkumar S[2] , Gouri Nandana C P[3] , Jeevitha S[4]

[1]Assistant Professor, Dept of CSE, Erode Sengunthar Engineering College, Perundurai.

[2,3,4]Student, Dept of CSE, Erode Sengunthar Engineering College, Perundurai.

***ABSTRACT* - Efficient hospital management is essential for optimizing patient flow, especially in Outpatient Departments (OPDs) and inpatient admissions. This project aims to develop HospiTrack - A Cloud Based Hospital Management System that enhances patient queue management, monitors bed availability, and streamlines admissions. The system will provide real-time updates on OPD queues, doctor availability, room facilities, and ambulance services, ensuring efficient resource allocation and reduced waiting times. Additionally, it will integrate seamlessly with a city-wide healthcare network for better hospital coordination. Leveraging cloud-based technology, HospiTrack ensures secure data storage, accessibility, and scalability, ultimately improving hospital efficiency and patient care.**

KEYWORDS:* *Hospital Management System, Real-Time Queue Management, Spring Boot, React.js, Cloud Computing, Role-Based Access Control, Healthcare Automation, MySQL, Web Application, Patient Flow Optimization

I. INTRODUCTION

In the current era of digital transformation, the healthcare sector is increasingly relying on technology to streamline operations, improve patient outcomes, and ensure optimal resource utilization. Hospitals, being complex and high-demand environments, face numerous challenges in managing daily workflows, particularly in outpatient departments (OPD), where long queues, inefficient tracking of patient status, and lack of centralized information often result in delayed treatments and patient dissatisfaction. The traditional manual systems used in many hospitals are not only time-consuming but also prone to human errors, making it difficult to maintain real-time visibility across different departments. With the growing expectations for faster service and increased transparency, there is a pressing need for integrated systems that offer real-time monitoring, smart automation, and secure access control. This is where modern web technologies can play a transformative role. The emergence of full-stack development frameworks has enabled the creation of scalable and responsive applications that can be tailored to meet the specific operational requirements of healthcare institutions. To address these challenges, this paper presents **HospiTrack**, a cloud-based hospital management system designed to simplify and automate critical hospital functions through a modular and real-time architecture. HospiTrack combines a user-friendly React.js frontend, a secure and efficient Spring Boot backend, and a robust MySQL database to provide an end-to-end solution for hospital administration. The platform incorporates key modules such as patient registration, OPD queue tracking, doctor scheduling, bed availability, ambulance service management, and emergency alert monitoring. A standout feature of the system is its real-time queue management functionality, which allows hospital staff to update patient consultation statuses dynamically while enabling patients to track their position in the queue without needing to log in. The system also includes a role-based access control mechanism, ensuring that sensitive data can only be accessed or modified

by authorized personnel. Furthermore, its cloud-ready design makes it highly accessible and deployable across different environments, from urban hospitals to remote clinics. The goal of this work is to demonstrate how combining modern web development tools with thoughtful system design can lead to a comprehensive healthcare management platform. The proposed solution not only enhances the efficiency of hospital workflows but also improves the overall experience for both administrators and patients. Future enhancements can include artificial intelligence for predictive analysis, mobile app support, and integration with IoT-based health monitoring devices to further elevate the system's capabilities.

II. LITERATURE SURVEY

2.1. "BED ALLOCATION OPTIMIZATION BASED ON SURVIVAL ANALYSIS, TREATMENT TRAJECTORY AND COSTS ESTIMATIONS" [1]

Effective bed allocation in intensive care units (ICUs) remains a critical challenge due to fluctuating patient demands, limited capacity, and the need for efficient resource management. Traditional approaches have largely focused on operational aspects such as queue modeling and demand forecasting. Early models addressed occupancy and patient flow but lacked integration of clinical parameters like survival rates and treatment trajectories. Recent research has incorporated machine learning and survival analysis into healthcare decision-making. Predictive models, such as those using logistic regression and survival estimators like Kaplan-Meier, have been employed to improve individual care planning. However, these models often function in isolation and are not integrated into optimization frameworks for bed assignment. Optimization techniques, particularly genetic algorithms, have shown promise in healthcare scheduling and resource allocation. However, prior studies have seldom combined clinical, economic, and trajectory-based insights into a unified model. The reviewed paper advances this field by integrating survival probabilities, cost estimations, and personalized treatment pathways as weighted constraints within a genetic algorithm. This multi-dimensional approach offers a balanced and dynamic solution to the complex problem of ICU bed allocation.

2.2. "A PARALLEL PATIENT TREATMENT TIME PREDICTION ALGORITHM AND ITS APPLICATIONS IN HOSPITAL QUEUING-RECOMMENDATION IN A BIG DATA ENVIRONMENT" [2]

This research introduces a Parallel Patient Treatment Time Prediction (PTTP) algorithm aimed at optimizing hospital queue management using real-time patient data within a big data framework. Long waiting times and overcrowded hospital queues are known to degrade the patient experience and resource efficiency. The PTTP algorithm addresses this by predicting treatment time for each patient task, leveraging historical healthcare data and cloud computing technologies. The model predicts the waiting time for each patient based on the treatment duration of patients ahead in the queue. These predictions are powered by realistic, large-scale hospital datasets. The algorithm operates in tandem with a Hospital Queuing-Recommendation (HQR) system, which suggests an optimized and personalized treatment plan for each patient. This recommendation system enables patients to access their estimated wait times via a mobile interface, thereby facilitating better time management and reducing stress. To meet the need for low-latency and high-efficiency, the solution is implemented using Apache Spark on a high-performance computing environment at the National Supercomputing Center in Changsha. Experimental results show that the proposed system can significantly reduce patient wait times and improve operational flow, making it highly applicable in modern hospital environments.

2.3. "GENETIC ALGORITHM BASED AUTOMATIC OUT-PATIENT EXPERIENCE MANAGEMENT SYSTEM (GAPEM) USING RFIDs AND SENSORS" [3]

This paper presents a novel framework for automating patient experience evaluation in hospitals by integrating RFID technology, automated feedback forms, Hospital Management Information System (HMIS) data, and environmental sensor inputs. The system, named GAPEM

(Genetic Algorithm-based Patient Experience Management System), uses Genetic Algorithms (GA) to optimize the computation of patient satisfaction scores, termed the Overall Satisfaction Index (OSI). The RFID tags track time spent by patients at each station, while feedback forms provide subjective satisfaction ratings. HMIS contributes structural data including staff, departments, and room information. Environmental sensors add context through real-time temperature and humidity readings. The GA processes this multifaceted input to derive optimized weight values, which are then used to compute a Patient Experience Management Index (PEMI) that closely matches the OSI. The approach aims to eliminate the need for manual statistics and enables real-time monitoring of patient satisfaction, allowing hospital staff to promptly address areas of concern. Experiments conducted in a local hospital using the prototype tool yielded an accuracy of 80.3%, validating the model's effectiveness in replicating human-generated satisfaction indices. This system allows healthcare providers to take proactive, data-driven actions to improve outpatient experiences.

III. PROBLEM STATEMENT

In many hospitals and healthcare facilities, managing patient flow, outpatient queues, and real-time resource availability remains a significant challenge. Traditional systems often rely on manual record-keeping or outdated software that lacks real-time synchronization and scalability. This leads to long waiting times, inefficient admission processes, poor visibility into bed occupancy, and delays in patient care — ultimately impacting hospital efficiency and patient satisfaction. Moreover, most existing solutions are not designed with dynamic queue tracking or real-time updates in mind, nor do they provide a clear distinction in access between administrators and general users. This limits transparency for patients and puts an unnecessary administrative burden on hospital staff, who must manually update records or provide repetitive status information. HospiTrack is designed to address these inefficiencies through a cloud-based, real-time hospital management system. It aims to streamline patient queue management, monitor bed availability, manage outpatient department (OPD) appointments, and handle emergency alerts through a centralized platform. With features such as real-time data updates, role-based

IV. SOLUTION STATEMENT

To address the inefficiencies in hospital operations and patient flow, HospiTrack proposes a comprehensive, cloud-based hospital management system designed to simplify and optimize everyday healthcare workflows. The system will offer a real-time, digital platform for managing outpatient queues, monitoring bed availability, updating patient status, and streamlining hospital admissions — all in one place. Built with a robust backend powered by Spring Boot and a modern, responsive frontend developed using React, the solution provides seamless communication between different hospital departments. Real-time data updates ensure that hospital staff can add, edit, or track patient records instantly, while patients and visitors can stay informed without needing to log in. The platform will include a clear distinction between user roles — allowing only admins (hospital staff) to modify records, while general users will have view-only access for transparency and convenience. Additionally, the system will feature modules for doctor scheduling, ambulance availability, blood bank status, and emergency alerts — all integrated into a single, intuitive dashboard. Notifications will alert staff to critical updates such as incoming emergencies or bed shortages, enabling quicker response times and better coordination. By hosting HospiTrack on a cloud platform such as AWS, Google Cloud, or Railway (free-tier), the application ensures high availability, scalability, and accessibility from any location. This solution empowers hospitals to reduce administrative workload, improve patient experience, and make data-driven decisions — ultimately transforming traditional healthcare administration into a smart, responsive digital experience.

V. METHODOLOGY

The development of HospiTrack follows a structured, step-by-step methodology designed to ensure the system is scalable, secure, user-friendly, and ready for real-world hospital environments. Each phase of the project is carefully planned and executed, focusing on aligning technology with real healthcare needs. The methodology combines best practices from full-stack development with

agile principles for iterative improvement and feature enhancement.

A. REQUIREMENT GATHERING AND ANALYSIS

The first step involved identifying the real-world challenges faced by hospitals, especially in managing outpatient queues, bed availability, and patient records. Feedback was gathered from hospital staff and healthcare professionals to understand their daily pain points. Based on this, key functional requirements like real-time updates, queue management, emergency alerting, and admin-controlled access were defined. A rough project plan and system scope were also created during this phase.

B. SYSTEM DESIGN AND TECHNOLOGY STACK

Once the requirements were clear, the system architecture was designed using a client-server model. The backend was planned with Spring Boot for API development and database interaction, while the frontend was structured using React JS for a responsive and user-friendly interface. A MySQL database was chosen to store structured data like patient records, queue entries, bed status, and emergency cases. Diagrams such as ER models and flowcharts were created to visualize system flow.

C. DEVELOPMENT OF FRONTEND AND BACKEND MODULES

The application was developed in a modular way. On the frontend, React components were created for pages like Login, Dashboard, Queue View, and Patient Form. React Router was used for navigation, and conditional rendering handled role-based visibility. On the backend, Spring Boot was used to build RESTful APIs for each module — Patient, Bed, Ambulance, Blood Bank, and Critical Alerts. JPA was used for database operations, and each service was properly layered using controller, service, and repository structures.

D. REAL-TIME UPDATES AND SECURITY MECHANISMS

To keep the system dynamic and responsive, real-time data updates were integrated through WebSockets and Polling. Whenever the queue or status was updated by the admin, it reflected instantly on the user side without requiring a page refresh.Spring Security is used to implement role-based authentication, where admins can log in and manage data while patients can view public information without login. Data access is secured using encrypted tokens and validation logic.

E. TESTING, DEPLOYMENT, AND DOCUMENTATION

After the development phase, the application was rigorously tested to ensure stability and correctness. Backend modules were verified through unit testing, especially the Spring Boot REST APIs, while the frontend was tested manually to validate UI interactions, form inputs, and real-time updates on the queue and dashboard. Once the system passed all functional and performance checks, it was deployed using Railway, a cloud-based platform well-suited for Spring Boot applications. Railway simplified the deployment by allowing seamless integration with GitHub, automatic builds, and direct database hosting using MySQL. This ensured the application was accessible over the internet with minimal setup. Lastly, comprehensive technical documentation and user guides were prepared to help hospital administrators and staff understand the usage, manage patient flow, and adopt the system efficiently.

VI. MODULES

The HospiTrack system is organized into a set of well-defined modules, each designed to address a specific aspect of hospital operations. These modules work in coordination to ensure a streamlined, efficient, and real-time hospital management experience. The key modules developed in the system are as follows:

A. PATIENT MANAGEMENT MODULE

This module is responsible for handling all patient-related data. It facilitates the registration of new patients, viewing of existing records, and updating of health status or consultation details. The system supports both outpatient (OPD) and inpatient (IPD) workflows and ensures that patient records are securely stored and accessible only to authorized personnel.

B. QUEUE MANAGEMENT MODULE

To minimize patient waiting time and enhance service delivery, this module provides a dynamic and interactive

queue system. It displays token numbers, patient names, and consultation statuses such as *Waiting, In Consultation, Consulted, and Cancelled*. Real-time updates are enabled through backend integration, and only users with administrative access are permitted to modify queue data, while public viewers can view the information of status and token number without authentication.

C. DOCTOR SCHEDULING AND MANAGEMENT

This module facilitates streamlined management of doctor profiles and their availability schedules. In the current system, general users can view essential information such as the doctor's name, area of specialization, available consultation timings, and working days. This transparency helps patients make informed decisions before visiting the hospital. On the other hand, administrative users are granted the authority to update and manage doctor availability in real time. They can modify consultation timings and active days dynamically based on operational needs or emergencies. This dual-access structure ensures that schedule information remains accurate and up-to-date while preserving access control integrity. By centralizing doctor schedules and enabling real-time updates, the system improves coordination between hospital staff and enhances the overall patient experience.

D. BED AND ROOM ALLOCATION MODULE

Designed to improve bed utilization, this module maintains real-time data on bed occupancy across different wards. It supports tracking of total, available, and occupied beds, and plays a crucial role in assigning beds to admitted patients. The integration with emergency and patient modules allows for quick decision-making during critical situations.

E. AMBULANCE SERVICES MANAGEMENT

This module provides real-time visibility into the hospital's ambulance fleet and their operational status. In the current implementation, users can view essential details such as the ambulance number, driver's name, contact information, current status (Available, On Duty, or Under Maintenance), present location, and assigned destination. This ensures transparency and helps hospital staff and patients stay informed about ambulance deployment. Administrative users are authorized to update critical details such as the vehicle's status, live location, and destination in real time, allowing for effective coordination and rapid response during emergencies. To enhance this functionality further, future iterations of the system aim to integrate live GPS tracking, enabling precise location monitoring of ambulances en route. This would significantly improve the efficiency of emergency handling and patient transfer operations.

F. BLOOD BANK INVENTORY MANAGEMENT

This module is designed to provide hospital staff and users with accurate, real-time information on the availability of various blood groups. The system allows users to check whether a specific blood type is currently available in the hospital, enabling faster access during emergencies or planned treatments. It supports real-time inventory updates and assists administrative users in monitoring stock levels and updating availability as needed.

G. EMERGENCY ALERT AND CRITICAL CASE HANDLER

To effectively manage life-threatening situations, this module enables authorized users to raise critical alerts. These alerts are immediately visible on the dashboard, and administrators are notified in real time. The system ensures prioritization of critical patients by flagging their cases within the queue and notifying relevant staff.

H. AUTHENTICATION AND ROLE-BASED ACCESS CONTROL

Security and access control are enforced through this module. Admin-level users can log in to modify hospital data, while general users (patients, visitors) can view publicly accessible information such as the queue or doctor availability. Spring Security is employed to enforce secure authentication and role-based permissions.

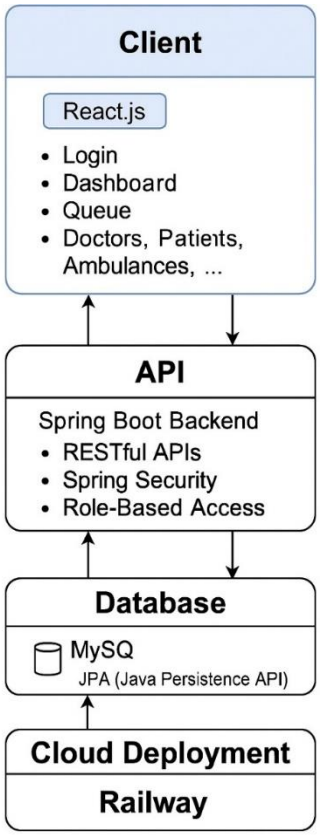
I. CENTRALIZED DASHBOARD AND MONITORING

The dashboard provides a holistic view of hospital operations by aggregating live statistics from all core modules. It displays key indicators such as total patient count, available beds, active queues, emergency alerts, and more. This centralized view helps hospital staff

monitor, assess, and act promptly based on real-time data insights.

VII. SYSTEM ARCHITECTURE

The architecture of the HospiTrack system follows a layered and modular design, ensuring maintainability, scalability, and secure interaction between components. It is based on a client-server model, with a React.js frontend, Spring Boot backend, and MySQL database. Real-time updates and secure access control are implemented using Spring Security and role-based authentication. The system is cloud-deployable and designed with future enhancements like live GPS tracking and blood donation integration in mind.



VIII. FUTURE ENHANCEMENTS

While the current version of HospiTrack meets essential hospital management needs, several enhancements are planned to improve system functionality, user experience, and operational efficiency. These future upgrades aim to expand the platform's capabilities and adaptability in real-world healthcare scenarios.

A. LIVE GPS AMBULANCE TRACKING

Integration of live GPS tracking will enable real-time monitoring of ambulance locations. This will enhance emergency response coordination, allowing hospital staff to track arrival times and optimize dispatch.

B. BLOOD DONATION MANAGEMENT SYSTEM

A module for managing voluntary blood donations can be integrated. It will allow users to register as donors, schedule donations, and receive alerts when their blood type is in demand, helping maintain optimal inventory levels.

C. DOCTOR APPOINTMENT BOOKING SYSTEM

Enabling patients to book appointments based on doctor availability can reduce queue congestion and waiting times. This feature will automate scheduling and allow patients to select preferred time slots.

D. MOBILE APPLICATION DEVELOPMENT

A cross-platform mobile application (Android/iOS) can make HospiTrack more accessible for both staff and patients, offering real-time updates, push notifications, and offline support for emergencies.

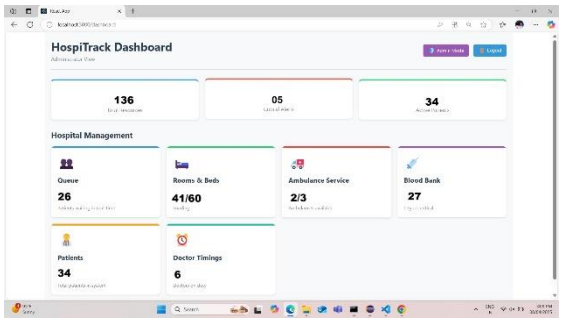
E. AI-BASED PREDICTIVE ANALYTICS

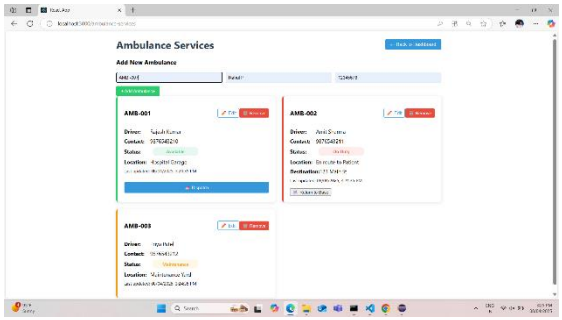
Leveraging AI/ML for predictive analytics can help forecast patient inflow, resource utilization, and emergency patterns. This will assist in proactive planning and enhance decision-making for hospital administrators.

XI. CONCLUSION

This paper presented **HospiTrack**, a smart, cloud-based hospital management system designed to streamline outpatient department (OPD) workflows, monitor real-time patient queues, manage bed availability, and enhance overall hospital resource coordination. By integrating a React-based frontend with a secure Spring Boot and MySQL backend, the system ensures seamless real-time data updates and role-based access control. The solution prioritizes both usability and scalability, enabling hospital staff to efficiently manage patient flow while allowing

patients to view their status without requiring login credentials. Our implementation demonstrates how modern web technologies can be effectively applied to digitize hospital operations, improving patient experience and operational efficiency. Future enhancements will include AI-driven analytics for resource forecasting, integration with live gps ambulance tracking and blood donation management system.





REFERENCES

[1] S. Safdar, S. A. Khan, A. Shaukat, and M. U. Akram, "Genetic Algorithm Based Automatic Out-Patient Experience Management System (GAPEM) Using RFIDs and Sensors," *IEEE Access*, vol. 7, pp. 78684–78696, 2019.

[2]J. Chen, K. Li, Z. Tang, K. Bilal, and K. Li, "A Parallel Patient Treatment Time Prediction Algorithm and Its Applications in Hospital Queuing-Recommendation in a Big Data Environment," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 6, pp. 2627–2636, Jun. 2018.

[3] A. Pfohl and C. Wuppermann, "Hospital Logistics: An Overview of Concepts and Applications," *International Journal of Healthcare Technology and Management*, vol. 10, no. 1/2, pp. 1–13, 2009.

[4] M. A. Badawi et al., "Real-time Healthcare Monitoring System for Early Detection of Heart Diseases Using IoT," *IEEE Access*, vol. 8, pp. 123424–123433, 2020.

[5] R. Agrawal, M. Kumar, and R. S. Tomar, "A Real Time Cloud Based Hospital Management System," *Procedia Computer Science*, vol. 70, pp. 208–222, 2015.

[6] A. Rahman, S. Arif, and A. A. Bakar, "Real-time Queue Monitoring System for Healthcare Services Using IoT," in *Proc. IEEE International Conference on Computer and Information Sciences (ICCIS)*, Kuala Lumpur, Malaysia, 2018, pp. 1–6.

[7] S. S. Sheikh and R. A. Shah, "Design and Implementation of Hospital Management System Using Blockchain and IoT," in *Proc. 2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE)*, Vellore, India, 2020, pp. 1–5.

[8] M. A. Ferrag, L. Maglaras, and H. Janicke, "Blockchain and IoT-Based Smart Healthcare System: A Survey," in *IEEE Access*, vol. 7, pp. 131761–131788, 2019.

[9] S. A. Hussain, "Electronic Medical Record Implementation: A Review," *International Journal of Engineering and Advanced Technology*, vol. 9, no. 3, pp. 112–115, Feb. 2020.

[10] A. N. K. Reddy and B. Bhavani, "Cloud Based Real Time Hospital Management System," *International Journal of Computer Applications*, vol. 139, no. 5, pp. 28–33, Apr. 2016.