IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

SMART ICU PATIENT MONITORING SYSTEM: A REAL-TIME IOT-BASED HEALTHCARE SOLUTION

[1]Dr. R. A. Burange, [2]Shreyash Almast, [3]Abhay Shivhare, [4]Nidhi Joshi

[1]Asst. Professor, [2]Student, [3]Student, [4]Student

[1]Electronics & Telecommunication Engineering,

[1]K.D.K. College of Engineering, Nagpur, India

Abstract: The **Smart ICU Patient Monitoring System** is a cutting-edge healthcare solution designed to continuously track and monitor vital signs of critically ill patients in intensive care units. This system integrates **IoT-based sensors** with a **Flask web application**, enabling real-time data acquisition and visualization. The system captures key health parameters, including **ECG, heart rate, oxygen saturation, and body temperature**, and transmits them to a **centralized dashboard** for medical staff. Critical threshold breaches trigger **instant alerts via SMS and voice calls**, ensuring timely medical intervention. The system utilizes **Raspberry Pi** as the primary processing unit and **SQLite** for efficient data management. The web application provides **secure authentication, real-time graphs, printable patient reports, and historical data access**. This project highlights the potential of IoT-based **remote patient monitoring**, ensuring enhanced healthcare services with improved efficiency and accuracy.

I. INTRODUCTION

The **Smart ICU Patient Monitoring System** is an advanced, real-time monitoring solution designed to track critical vital signs of patients in intensive care units (ICUs). The system captures real-time sensor data, processes it using a **Raspberry Pi**, and transmits the information to a **Flask-based web application**. The web interface displays vital signs, generates alerts for abnormal readings, and provides real-time access to patient data for healthcare professionals.

II. OBJECTIVES

- To design a system capable of continuously monitoring patient vitals such as **heart rate, ECG, oxygen saturation, and body temperature**.
- To implement **real-time data visualization** on a professional and interactive **web dashboard**.
- To develop an **alert system** that notifies healthcare providers via **SMS and voice call** in case of critical conditions.
- To ensure secure **user authentication** and **database management** for storing patient details and vital signs.

III. APPLICATIONS

- **Hospitals & ICUs:** Real-time patient monitoring to detect emergencies and alert medical staff.
- **Remote Healthcare:** Can be adapted for telemedicine and home-based patient monitoring.
- **Elderly Care Centers:** Continuous health monitoring for elderly individuals at risk of cardiac or respiratory issues.
- **Military & Disaster Relief Camps:** Monitoring of patients in remote locations where immediate medical attention is necessary.
- **Research & Development:** Data collection for medical studies and AI-based health predictions

IV. Technologies Used

Hardware Components:

- Raspberry Pi (Main Processing Unit)
- ECG Sensor
- Temperature Sensor
- Pulse Oximeter Sensor
- LCD/3.5-inch Touchscreen Display

Software & Tools:

- Flask (Web Framework)
- SQLite (Database Management)
- Twilio API (SMS & Voice Call Alerts)
- HTML, CSS, JavaScript (Frontend Development)
- Python (Backend Development & Sensor Integration)

V. Project Implementation:

Step 1: Web Application Development

- Designed and developed a Flask-based web interface with an intuitive dashboard displaying real-time patient data.
- Implemented user authentication to ensure only authorized personnel can access the dashboard.
- Integrated a printable report generation feature for patient records.

Step 2: Database Setup

- Used SQLite to manage patient details and historical vital sign data.
- Established a secure database initialization within the main application file.

Step 3: Sensor Integration with Raspberry Pi

- Collected real-time data from sensors connected to the Raspberry Pi.
- Developed Python scripts to process sensor data and send it to the Flask server.
- Tested each sensor individually before full integration.

Step 4: Alert System Implementation

- Implemented real-time alerts for critical conditions using Twilio API.
- Tested SMS and voice call notifications for threshold breaches in vitals.
- Designed visual alerts on the web dashboard for immediate attention.

Step 5: Final Testing and Deployment

- Conducted end-to-end testing to ensure seamless integration of software and hardware.
- Performed stress tests to check system reliability under different conditions.
- Successfully deployed the project for demonstration and validation.

VI. Challenges Faced & Solutions:

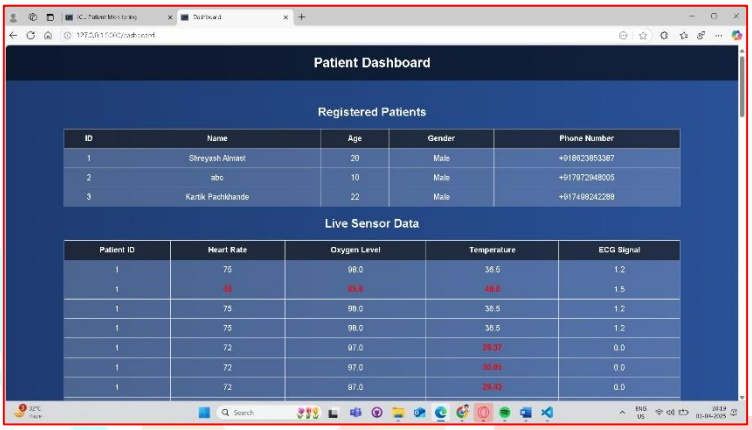
Table 1: Challenges and Solution

Challenges	Solutions Implemented
Sensor Calibration Issues	Performed multiple iterations of testing and correction.
Integration of Multiple Sensors	Developed separate scripts for each sensor before merging.
Web Dashboard Performance	Optimized API calls and data retrieval methods.
Alert System Testing	Used simulated data before real sensor integration.

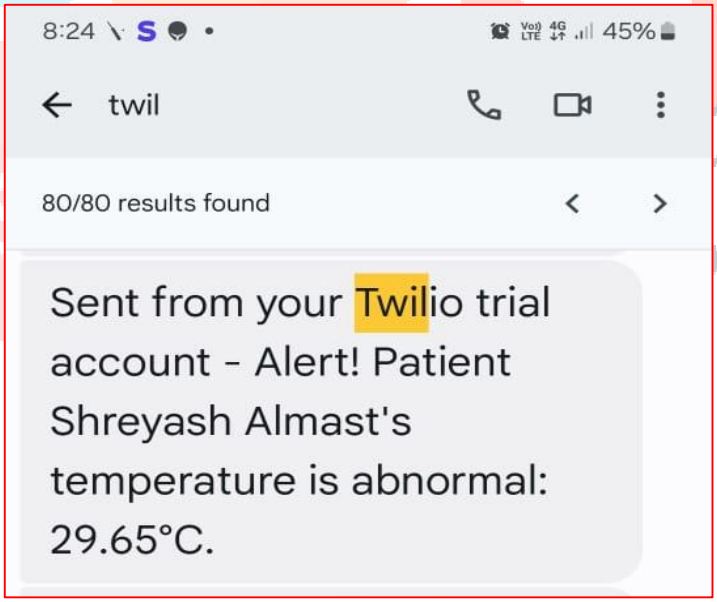
VII. OUTPUT:

The Smart ICU Patient Monitoring System provides the following outputs:

- Live Monitoring Dashboard: Displays real-time vital signs data with an intuitive graphical interface.



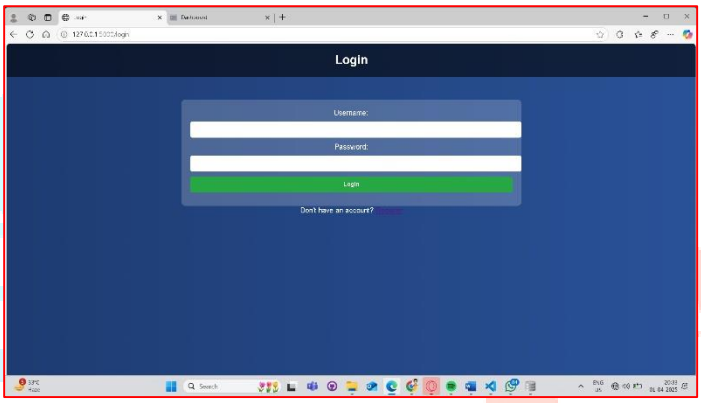
- Alert Notifications: Immediate SMS and voice call alerts for abnormal conditions.



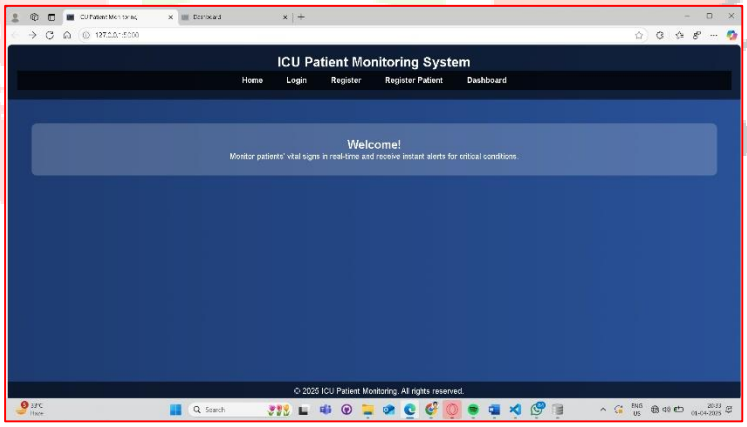
- Patient Reports: Printable patient reports containing historical health data.

id	patient_id	heart_rate	oxygen_level	temperature	ecg_signal
Fil...	Filter	Filter	Filter	Filter	Filter
1	1	75	98.0	36.5	1.2
2	1	45	85.0	40.0	1.5
3	1	75	98.0	36.5	1.2
4	1	75	98.0	36.5	1.2
5	1	72	97.0	29.37	0.0
6	1	72	97.0	30.05	0.0
7	1	72	97.0	29.43	0.0
8	1	72	97.0	32.85	0.0
9	1	72	97.0	32.41	0.0
10	1	72	97.0	33.39	0.0
11	1	72	97.0	29.29	0.0
12	1	72	97.0	29.23	0.0
13	1	72	97.0	29.45	0.0
14	1	72	97.0	29.63	0.0
15	3	72	97.0	29.23	0.0
16	3	72	97.0	29.15	0.0
17	3	72	97.0	29.05	0.0

- Secure Access: Restricted access through user authentication, ensuring data privacy.



- Responsive UI: User-friendly interface accessible across multiple devices.



VIII. CONCLUSION:

The Smart ICU Patient Monitoring System successfully meets its objectives by providing a real-time monitoring platform for ICU patients. The integration of sensors with a Flask web application ensures seamless data transmission and visualization. The implemented alert system enhances patient safety by notifying healthcare providers about abnormal vital signs. This project demonstrates the potential of IoT and cloud-based monitoring in modern healthcare.

IX . FUTURE ENHANCEMENTS

- Implement **AI-based health predictions** using machine learning for early disease detection.
- Develop a **mobile application** for remote monitoring and alert notifications.
- Integrate **cloud storage** to enable data access from multiple locations securely.
- Enhance security with **role-based access control (RBAC)** to restrict data visibility.
- Expand sensor integration by adding **blood pressure and respiratory rate sensors**.
- Introduce **voice-based alerts** for emergency notifications.

X. REFERENCES

[1] **Kumar, S., & Gupta, M. (2020).** "Real-Time Health Monitoring System Using Raspberry Pi." International Journal of Engineering Research and Applications, 10(5), 37-42.

[2] **Choudhary, A., & Rathi, S. (2020).** "Design and Implementation of a Raspberry Pi-Based ECG Monitoring System." International Journal of Advanced Research in Computer Science, 11(5), 15-20.

[3] **Mahmoud, M. A., & Mohammed, A. M. (2021).** "Smart Health Monitoring System Using Raspberry Pi and IoT." Journal of Health and Environmental Research, 7(2), 105-113.

[4] **Alam, F., & Sinha, A. (2019).** "IoT Based Patient Health Monitoring System Using Raspberry Pi." International Journal of Computer Applications, 182(25), 34-39.

[5] **Kumar, V., & Singh, P. (2021).** "IoT Based Health Monitoring System Using Raspberry Pi." International Journal of Scientific & Engineering Research, 12(4), 1001-1006.

[6] **Khan, S., & Khan, M. (2019).** "Remote Health Monitoring System Using Raspberry Pi and IoT." International Journal of Engineering and Advanced Technology, 8(6), 1-6.

[7] **Nagaraja, K. B., & Ramesh, G. (2020).** "Design of an Efficient Health Monitoring System Using Raspberry Pi." Journal of Health Informatics in Developing Countries, 14(1), 22-30.

[8] **Mohan, R. K., & Arora, A. (2021).** "Design of Wearable Health Monitoring System Using Raspberry Pi." International Journal of Research in Engineering and Science, 9(3), 10-14.