Project Proposal: Hospital Patient Monitoring and Bed Allocation System

Sanjna
23K-2022

Syed Waleed Hussain
23K-0885

Muhammad Huzaifa Altaf
23K-0627

Shayan Nemat
23K-0899

March 24, 2025

1 Introduction

In modern healthcare facilities, efficient patient monitoring and bed allocation are critical for smooth operations. Our project aims to develop a multithreaded hospital management system that ensures optimal patient flow by managing check-ins, discharges, and emergency intakes dynamically. The system will utilize threads, mutexes, and semaphores to handle concurrent requests efficiently while prioritizing emergency cases.

2 Objectives

- Develop a real-time patient monitoring system with efficient bed allocation.

- Implement thread synchronization using mutexes and semaphores to prevent data inconsistencies.

- Handle priority-based preemptions for emergency cases.

- Simulate hospital operations such as patient admission, discharge, and ICU bed management.

- Ensure data integrity and avoid race conditions in concurrent operations.

3 System Features

3.1 Patient Admission & Discharge

- Threads handle patient check-in and discharge.

- Mutex locks prevent race conditions in patient records.

3.2 Bed Allocation & Management

- Semaphores manage the number of available beds.

- Emergency cases preempt regular patient allocations.

3.3 Priority-Based Resource Allocation

- Emergency cases have higher priority.

- Preemption mechanism implemented using priority queues.

3.4 ICU & General Ward Handling

- Separate semaphores for ICU and general ward beds.

- Dynamic allocation based on patient severity.

3.5 Real-Time Monitoring & Logging

- Logs every patient's check-in, discharge, and emergency entry.

- Tracks bed occupancy status.

4 Technology Stack

- Programming Language: C/C++ (for multithreading and synchronization mechanisms)

- Threading & Synchronization: POSIX Threads (Pthreads), Mutexes, Semaphores

- Data Handling: Structs and dynamic memory allocation

- Logging & Output: File handling for record-keeping

5 Expected Challenges

- Implementing efficient thread synchronization.

- Managing priority inversion for emergency cases.

- Ensuring real-time updates with minimal processing delays.

6 Conclusion

This project will serve as a practical implementation of real-time hospital management, helping optimize bed allocation while handling emergency scenarios efficiently. Our solution will demonstrate the use of multithreading, synchronization, and priority-based resource allocation, making it a robust system suitable for real-world healthcare applications.