

**PROJECT REPORT**

**CPU SCHEDULING ON DOCTOR'S APPOINTMENT MANAGEMENT SYSTEM**

**Course: Operating System LAB**

**Submitted to: SIR AADRISH PIRZADO**

**Submitted by:**

**AMRAH IMTIAZ**

**MUHAMMAD TAHIR**

## **Contents**

**Introduction**[**:**](https://docs.google.com/document/d/1-cuJp7bguH9KUPfCa9_fu7waX9_NBTvbB-Jo10bP8-U/edit#heading=h.4d34og8)

**Objectives**[**:**](https://docs.google.com/document/d/1-cuJp7bguH9KUPfCa9_fu7waX9_NBTvbB-Jo10bP8-U/edit#heading=h.2s8eyo1)

**Scope**[:](https://docs.google.com/document/d/1-cuJp7bguH9KUPfCa9_fu7waX9_NBTvbB-Jo10bP8-U/edit#heading=h.17dp8vu)

Algorithm Implementation:[:](https://docs.google.com/document/d/1-cuJp7bguH9KUPfCa9_fu7waX9_NBTvbB-Jo10bP8-U/edit#heading=h.3rdcrjn)

User Interaction:[**:**](https://docs.google.com/document/d/1-cuJp7bguH9KUPfCa9_fu7waX9_NBTvbB-Jo10bP8-U/edit#heading=h.26in1rg)

Calculation and Output[:](https://docs.google.com/document/d/1-cuJp7bguH9KUPfCa9_fu7waX9_NBTvbB-Jo10bP8-U/edit#heading=h.lnxbz9)

Educational Utility:

Execution on Ubuntu Linux Terminal[:](https://docs.google.com/document/d/1-cuJp7bguH9KUPfCa9_fu7waX9_NBTvbB-Jo10bP8-U/edit#heading=h.lnxbz9)

**Methodology**[**:**](https://docs.google.com/document/d/1-cuJp7bguH9KUPfCa9_fu7waX9_NBTvbB-Jo10bP8-U/edit#heading=h.2s8eyo1)

Clarify Requirements:

Design Phase:

Implementation[:](https://docs.google.com/document/d/1-cuJp7bguH9KUPfCa9_fu7waX9_NBTvbB-Jo10bP8-U/edit#heading=h.lnxbz9)

Testing:

Documentation:

Code[**:**](https://docs.google.com/document/d/1-cuJp7bguH9KUPfCa9_fu7waX9_NBTvbB-Jo10bP8-U/edit#heading=h.4d34og8)

Output[**:**](https://docs.google.com/document/d/1-cuJp7bguH9KUPfCa9_fu7waX9_NBTvbB-Jo10bP8-U/edit#heading=h.4d34og8)

Conclusion[**:**](https://docs.google.com/document/d/1-cuJp7bguH9KUPfCa9_fu7waX9_NBTvbB-Jo10bP8-U/edit#heading=h.4d34og8)

**PROJECT PROPOSAL**

**1) Introduction:**

In modern healthcare, the efficient management of patients is important for the timely and effective administration of medical care. Health providers usually find difficulty scheduling and prioritizing doctor visits, delaying treatment, increasing waiting time, and leading to patient dissatisfaction.

Our project is an automated scheduling system, called the Doctor's Appointment System, where normal and emergency cases are handled effectively. Patients enter their personal information and categorize the case as either normal or emergency. Normal cases are scheduled using the First-Come, First-Served algorithm; thus, the order of processing is maintained. Pre-emption is used for emergency cases, where the priority of urgent cases guarantees to be attended to immediately.

Implemented in C++ and designed to execute in an Ubuntu Linux terminal, this system makes the management of patients more efficient, decreases the waiting times for emergency cases, and ensures that routine check-ups are carried out smoothly, thus improving patient satisfaction and healthcare outcomes.

**2) Objectives:**

**1. Automation of Doctor's Appointment:** Implement a system that automatically schedules doctors' appointments and facilitates a seamless working experience in the clinic.

**2. Emergency Cases:** Implement a priority-based preemptive algorithm to allow emergency appointments to be processed first, thus improving the response time to critical medical conditions.

**3. Efficiency:** Implement the First-Come, First-Served (FCFS) algorithm for normal appointments, which maintains an orderly and just scheduling system and lowers the administrative burden on staff.

**4. Patient Experience:** Reduce the waiting time and provide proper medical care, thereby improving patient satisfaction and confidence in the health system.

**5. Data Collection and Management:** To systematically and proficiently collect and manage patients' information, including personal data and health conditions, for accurate and secure management of data during the procedures of the study.

**6. System Implementation:** Implement this scheduling system using C++ to execute on Ubuntu Linux terminal environments for improved performance and compatibility with most healthcare IT infrastructures.

**7. Optimize Healthcare Delivery:** Help bring about improved patient health outcomes by making medical consultations and the use of healthcare resources as efficient as possible.

**3) Scope:**

The scope of this project, Doctor's Appointment System, is to design an automated scheduling solution using operating system concepts, specifically CPU scheduling algorithms, to manage and prioritize doctor appointments. The following goals and deliverables will be kept in focus:

**Goals:**

⦁ Automate the appointment scheduling process to enhance efficiency and reduce the workload of staff.

⦁ Implement FCFS and priority-based preemptive algorithms to handle normal and emergency cases, respectively.

⦁ Improve patient satisfaction by minimizing waiting times and ensuring timely medical care.

⦁ Ensure safe collection and management of patient data.

**Deliverables:**

⦁ A fully functional C++ program that executes in an Ubuntu Linux terminal environment.

⦁ An interface for patients to input personal information and categorize their medical condition.

⦁ A scheduling module that allots appointment IDs according to the type of appointment being normal or an emergency.

⦁ Documentation regarding the steps on how to execute this project in UBUNTU LINUX terminal.

**4) Methodology:**

**⦁Planning and Designing:**

The planning and design will start by trying to understand the requirements of the project, the Doctor's Appointment System, and the goals that must be achieved. The functionalities that will be used—patient data input, scheduling for appointments, and handling priorities—will be outlined.

**⦁Development:**

Start coding the system in C++. Begin with the user interface that allows for input of patients' personal information and categorization of appointments as either normal or emergency. Implement the scheduling algorithms—FCFS for normal appointments and priority-based preemptive for emergencies. The system should handle data in a secure and efficient way.

**⦁Testing:**

Testing will be done to ensure that the system works as expected and solve all bugs and issues faced.

**⦁Deployment Preparation:**

Make sure everything is set up for the system to be deployed on Ubuntu Linux. All the dependencies should be included, and it should work smoothly in the Linux environment. Write some basic documentation for users to know how to use the system.

**5) CODE:**

**6) OUTPUT**

**5) Conclusion:**

In conclusion, the Doctor's Appointment System project is not only a solution for scheduling challenges in healthcare but also a project for learning concepts of operating system (CPU Scheduling algorithms), C++ programming, and UBUNTU Linux. We have designed a system that streamlines appointments to reduce waiting times and guarantees timely care to the patient through automated scheduling algorithms and easy-to-use interfaces.