Module Code & Module Title

Level 5 – Operating Systems

Assessment Type

Logbook 1

Semester

2025/26 Spring/Autumn

Student Name: Prabin Pradhan

London Met ID: 24046428

College ID: np01nt4a240115

Assignment Due Date: Sunday, September 14, 2025

Assignment Submission Date: Tuesday, September 16, 2025

Submitted To: Sarika Dahal

Word Count (Where Required): 1041

I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded.

13% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Match Groups

- 4 Not Cited or Quoted 9%
  Matches with neither in-text citation nor quotation marks
- 4 Missing Quotations 4%
  Matches that are still very similar to source material
- 0 Missing Citation 0%
  Matches that have quotation marks, but no in-text citation
- 0 Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

0%	Internet sources
0%	Publications
13%	Submitted works (Student Papers)

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Submitted works	Southern New Hampshire University - Continuing Education on 2023-04-03	7%
2	Submitted works	National University of Ireland, Galway on 2022-12-13	2%
3	Submitted works	American InterContinental University on 2023-09-06	2%
4	Submitted works	Manipal University Jaipur Online on 2025-01-14	2%

Table of Contents

List of Figures

List of Tables

Process Management

1 Introduction:

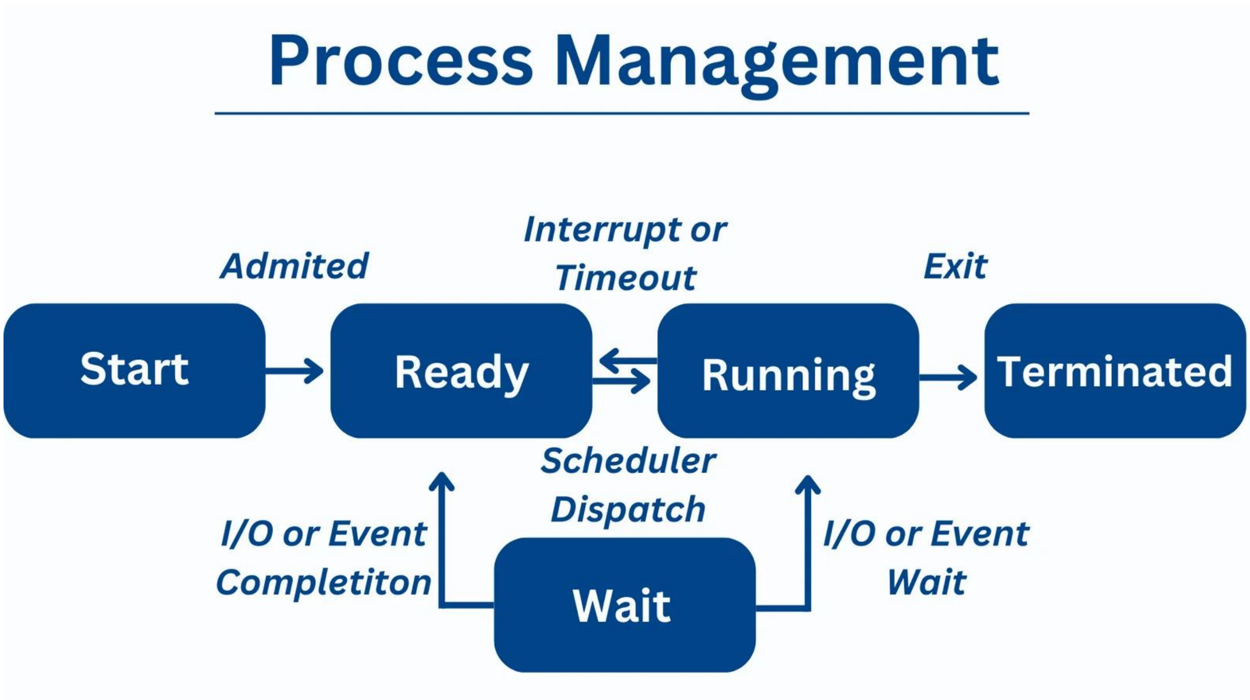

Figure 1: Process Management (Learn loner)

Process management in OS entails carrying out a variety of tasks, including creation of a process, scheduling, deadlocks, and process removal. The operating system is accountable for controlling all the continuous operations of the system. By undertaking activities like resource allocation and scheduling of a process, the operating system manages processes. When a process runs on a computer, the machine's CPU and RAM are both engaged. Additionally, the operating system must synchronize the many computer activities (Sharma, 2022).

What is a Process?

A process, otherwise known as a thread, is a processing unit of a program that performs some operation (Sharma, 2022).

A process is a running programmed. While we construct a C or C++ programmed, for instance, the compiler forms binary code. Raw and binary codes are programmed.

When we run the binary code, it turns into a process. A procedure is divided into four segments (Sharma, 2022):

Text: The present actions are expressed by the value of the text (Sharma, 2022).

Stack: Local variables, function parameters, return addresses, and other temporary data are stored on the Counter Stack (Sharma, 2022).

Storage: The global variables are stored in data (Sharma, 2022).

Heap: Dynamically allocated memory that is operated on at runtime (Sharma, 2022).

Process Control Block is an operating system control block to manage a process (PCB). It's a data structure held independently by the operating system for each process. Each PCB may be allocated a Process ID of type integer, which helps store all the information needed to keep track of processes (Sharma, 2022).

The data of CPU registers, which are formed as a process transition from one state to another, can also be stored on a PCB (Sharma, 2022).

What is Process Management in OS?

There are multiple processes in the process management system that use a common shared resource. Hence, the operating system needs to efficiently and effectively manage all operations and structures (Sharma, 2022).

In order to keep the system consistent, some things can have to be performed by one operation at a time. Otherwise, the system can be inconsistent and a deadlock will be encountered (Sharma, 2022).

In Process Management, the following activities are managed by the operating system (Sharma, 2022).

- Scheduling of process and thread on the CPU (Sharma, 2022).
- Both user processes and system processes can be created and deleted (Sharma, 2022).
- Suspend and resume processes (Sharma, 2022).
- Providing synchronization methods for processes (Sharma, 2022).

2 Objective:

The Process management in operating systems has several key objectives (TecnoDigital, 2025)

- Maximize system resource utilization_: The operating system must efficiently allocate system resources like CPU and memory to maximize utilization and avoid bottlenecks (TecnoDigital, 2025).

- Provide an immediate and soft response: The operating system must ensure that interactive processes, such as user programs, receive an immediate response to enable an improved user experience (TecnoDigital, 2025).

- Equity in resource allocation: The operating system must distribute resources equally among the processes so that no single process uses all the resources and affects other processes negatively (TecnoDigital, 2025).

- Maximize system-wide performance: Excellent process management would result in optimum performance of the system as a whole, minimizing wait time and maximizing efficiency (TecnoDigital, 2025).

3 Types of Process Management in an Operating System:

The OS handles various types of processes depending on system design (Aiman, 2025):

1 User Processes:

These are initiated by users. For examples web browser, games and require interaction or direct control (Aiman, 2025).

2 System Processes

These are initiated by the OS itself and help run core system services like memory management, I/O control, etc. (Aiman, 2025).

3 Foreground vs Background Processes

The following shows different between of Foreground and Background Processes in the table.

Table 1: Different between Foreground and Background.

Foreground Processes	Background Processes
Foreground services are those that require user interaction or are directly visible and active on the user interface (UI) (Parker, 2024).	Background services operate without direct user interaction and often run continuously or periodically to perform specific tasks (Parker, 2024).
Foreground services are visible to users and require active user interaction (Parker, 2024).	Background services operate invisibly or with minimal user interface (Parker, 2024).
Foreground services typically receive higher priority in resource allocation, ensuring responsiveness to user commands (Parker, 2024).	while background services run efficiently in the background (Parker, 2024).
Foreground services often handle complex, interactive tasks (Parker, 2024).	whereas background services focus on automated, repetitive tasks without user intervention (Parker, 2024).

4 Inter-Process Communication

Often, processes need to communicate and share data. IPC is a mechanism that allows this interaction, essential in distributed systems and multitasking environments (Aiman, 2025).

Common IPC Techniques:

Table 2: IPC Techniques.

Shared Memory	Multiple processes access a common memory space (Aiman, 2025).
Message Passing	Processes send and receive structured messages (Aiman, 2025).
Pipes	Unidirectional or bidirectional data flow between processes (Aiman, 2025).
Sockets	Network-based communication between processes (local or remote) (Aiman, 2025).

These methods ensure coordination between processes without conflicts, which is crucial for maintaining system integrity (Aiman, 2025).

4 Conclusion:

This research helps me to understand Process management, which is a vital component of modern operating systems, which makes multitasking seamless, sharing resources efficient, and system performance solid. Through processes like process scheduling, synchronization, and inter-process communication, the OS enables both user processes and system processes to operate in coordination without contention or deadlock.

By demarcating procedures as user/system and foreground/background types, the OS can schedule operations based on urgency and level of interaction. Shared memory, passing messages, pipes, and sockets are other IPC mechanisms which further allow processes to coordinate and exchange data, especially in distributed and concurrent environments.

The objectives of process management such as optimal utilization of resources, ensuring responsiveness, equity, and overall optimization of performance are achieved employing strategic scheduling algorithms and accurate process state management. The OS manages and tracks each process effectively employing Process Control Blocks (PCBs) without compromising the stability and integrity of the system.

In essence, effective process management transforms a computer system from a passive to an active machine that is dynamic, responsive, and intelligent enough to manage complex workloads. As computing needs continuously evolve, it remains important for system designers, administrators, and developers to become proficient in the art of process management.

References

Scaler. 2022. *Process Management in Operating System*. Apurva Sharma. Scaler. Available at: https://www.scaler.com/topics/process-management-in-operating-system/#what-is-process-management-in-os-

(Accessed 14 September 2025).

TecnoDigital. 2024. *Process management in operating systems*. InformaTecDigital. Available at: https://informatecdigital.com/en/process-management-in-operating-systems/

(Accessed 14 September 2025).

Aiman, J. 2025. *Process Management in Operating System*. MindScribes. Available at: https://mindscribes.com/process-management-in-operating-system/

(Accessed 14 September 2025).

Parker, J.W.S. 2024. *What is the difference between a foreground and a background service?* James Parker. Available at: https://www.jamesparker.dev/what-is-the-difference-between-a-foreground-and-a-background-service/

(Accessed 14 September 2025).