LESSON 68

Operating Systems Layers

80 minutes (2 periods) | Computer Systems and Networks | Outcome 2.11 | Outcome 1.15 | Outcome 1.16 | Outcome 1.12 | Outcome 2.1 | Outcome 1.20 | Outcome 1.23

WHAT THIS LESSON DOES

In this lesson, you'll explore the essential layers of an operating system and their roles in managing computer resources. Learn about hardware, kernel, application, and user interface layers, and understand process management, file systems, and virtualisation through practical examples.

CURRICULUM OUTCOMES

Outcome	What the specification asks for
2.11	describe the different components within a computer and the function of those components
1.15	consider the quality of the user experience when interacting with computers and list the principles of universal design, including the role of a user interface and the factors that contribute to its usability
1.16	compare two different user interfaces and identify different design decisions that shape the user experience
1.12	compare the positive and negative impacts of computing on culture and society
2.1	use abstraction to describe systems and to explain the relationship between wholes and parts
1.20	collaborate and assign roles and responsibilities within a team to tackle a computing task
1.23	reflect and communicate on the design and development process

WHAT STUDENTS SHOULD BE ABLE TO DO

- Understand the primary layers of an operating system and their specific functions in managing computer resources.
- Explain the concepts of process management, file systems, and virtualisation using practical examples.
- Analyse how operating system layers interact to ensure efficient and user-friendly computing.
- Apply knowledge of operating system components to real-world scenarios and tasks.
- Develop critical thinking through group discussions on operating system features and their applications.

BEFORE THE LESSON

- This is a theory and discussion lesson, so no editor or install is required. If you plan to demonstrate live, open Task Manager on Windows or run 'top' or a terminal on a Linux machine so students see real processes.
- Have two or three scenario prompts ready on the board for the group activity in case groups stall (running multiple phone apps, managing files on a school PC, setting up a VM for testing).

Operating Systems Layers

HOW THE LESSON RUNS

Introduction
Understanding OS Layers
Opening a Web Browser
Process Management
File Systems
Virtualisation
Group Activity: Analysing OS Features
Wrap Up

TEACHING MOVES

- 2. Understanding OS Layers. Draw the four layers as a stack on the board and keep it visible all lesson. Stress the direction of control: user interface at the top, hardware at the bottom, kernel doing the actual mediating between them.
- 3. Opening a Web Browser. Walk the browser example down the stack out loud: click the icon (user layer), the request passes to the kernel, which schedules a process and asks the hardware for memory. This makes the abstract layers concrete.
- 4. Process Management. Use the traffic-controller image, then open Task Manager or 'top' live so students see dozens of real processes competing. Point out that a process is a running program, not the program file on disk.
- 5. File Systems. Contrast NTFS on Windows with a Linux file system by name only. Emphasise that the file system is what stops data becoming chaos, and mention permissions and journaling as integrity features.
- 6. Virtualisation. Define hypervisor plainly before using it: software that shares one machine's real hardware across several virtual machines that each think they own it. Link this back to the kernel from the layer diagram.
- 7. Group Activity: Analysing OS Features. Ask each group to name which layer or function is doing the work at each point in their scenario. Insist they use the vocabulary from the board rather than describing it in everyday terms.
- 8. Wrap Up. Cold-call the four layers and the three kernel functions rather than reading the takeaways aloud. Have students give one real example each.

WHAT TO WATCH FOR

- Students confuse a program with a process, thinking the file on disk and the running instance are the same thing. Show two windows of the same browser open in Task Manager or 'top'. One program, several processes. Ask which one uses CPU and memory: the running process, not the file.
- Students think a virtual machine is a separate physical computer or an emulator running online, rather than a software environment sharing the one machine's real hardware. Return to the layer diagram. The hypervisor sits above one set of real hardware and hands each VM virtual CPU, memory and storage. There is still only one physical machine underneath.

DIFFERENTIATION

- Emerging: Give them just the layer diagram and ask them to label the browser example on it. Getting the four layers in the right order is the win before any of the kernel functions.
- Developing: Ask them to explain the difference between process management and file systems in their own words before joining the group activity, so they are reasoning rather than reciting.
- Proficient: Push toward Higher Level treatment: ask how the scheduler decides which process gets the CPU, or what journaling actually protects against. This analysis is strong evidence for an Applied Learning Task on systems.