



Student Engagement & Mentoring in Technology

1. “Build an OS” Interactive Simulation

Objective: Teach learners how operating systems manage core resources by letting them allocate system functions dynamically.

How It Works:

- A **drag-and-drop dashboard** allows learners to simulate managing:
 - Memory
 - Disk storage
 - Processes
 - Device drivers
 - User permissions
- Each system function (e.g., launching an app, adding a user, printing a file) creates demands on OS resources.
- The learner must correctly assign tasks (e.g., load driver, allocate RAM, terminate process) to keep the system running efficiently.
- Introduce “real-time” issues (e.g., malware attempt, frozen app, full disk) to challenge learners with troubleshooting.

Why It Works:

Encourages **applied understanding** of abstract concepts like memory management, process scheduling, and device handling — all through gamified logic.

2. “OS Match-Up Matrix” Sorting Activity

Objective: Help learners classify different OS types and their roles.

How It Works:

- Learners drag and match:
 - OS logos (e.g., Android, macOS, RTOS, Ubuntu Server)
 - Platform types (Mobile, Desktop, Server, Embedded)
 - Use cases (IoT device, personal laptop, web hosting)
- After matching, learners click on a match to **reveal a quick fact** explaining why that OS fits that category.

Why It Works:

Supports **pattern recognition** and **conceptual clarity**, making distinctions between OS types more intuitive and memorable.



Student Engagement & Mentoring in Technology

3. “Troubleshoot the Terminal” Text Adventure

Objective: Explore how OS process/application/device management works by solving common real-world problems.

How It Works:

- Learners enter a command-line style interface (simulated visually, not actual code).
- They navigate **troubleshooting scenarios** like:
 - “The printer won’t respond.”
 - “You’re running out of RAM.”
 - “A user forgot their login password.”
- Presented with command options like:
 - kill process
 - view running tasks
 - check user access
 - load driver
- Learners choose actions to fix the system, with feedback after each decision.

Why It Works:

Encourages **problem-solving** and **sequential reasoning** while introducing technical vocabulary and real administrative logic.

Summary Table

Interactive Activity	Focus Area	Learning Strategy
Build an OS Simulation	Core OS functions & resource management	Gamified simulation
OS Match-Up Matrix	OS types and usage classification	Visual pattern recognition
Troubleshoot the Terminal	Process, user, and device management scenarios	Branching scenario-based learning