

# Operating System Structure and Services Lecture 1 Notes

Posted in 10 Computer Science

The image shows a book cover for '10 Computer New Book 2026/2027 Operating System Structure and Services Lecture 1 Chapter 1'. The cover features a central diagram of the operating system structure with four levels: USER (white box with people icon), APPLICATIONS (green box with application icons), OPERATING SYSTEM (blue box with gear icon), and HARDWARE (purple box with hardware icons). To the right of the diagram is a list of 'OPERATING SYSTEM SERVICES' including User Interface, Program Execution, I/O Operations, File System Management, Communication, Error Detection, and Resource Allocation. At the bottom, there are links to the website 'SalmanBlogs.Com' and the YouTube channel 'ComputerWithSalman'.

These lecture notes are prepared by **Salman Ahmad**, who is both a **teacher** and a **web developer**. The purpose of this material is to make the topic clear and useful for students of Class 10 Computer Science.

## Table of Contents

- Introduction To Operating System
- Operating System as the Central Controller
- Role in user Hardware Interaction
- Responsibilities in Multi-User Environment Tasks
- Creating and Managing User Accounts
- Architecture of an Operating System
- Diagram or Structure

Kernal-VS-Shell

Types of Shell

OS Layers and Modular Design

MCQs

MCQs for testing

FAQs

## Introduction To Operating System (OS)

An Operating System (OS) is system software that manages computer hardware, runs applications, and provides a user interface.

### Examples:

- Microsoft Windows
- macOS
- Linux
- Android
- iOS

## Operating System as the Central Controller

The operating system works like a traffic controller of a computer. It manages:

- Which task should run first
- How memory should be used
- Which devices should work at a certain time

**the OS manages all these tasks together without crashing the system.**

## Role in user Hardware Interaction

Computer hardware cannot directly understand human language. The operating system acts as a bridge or translator between the user and the hardware.

### Example:

When you click an icon or type on a keyboard:

- Now the operating system converts your actions into machine instructions[0,1].
- Then the hardware performs the required task.

**This allows people to use computers easily without understanding hardware details.**

## Responsibilities in Multi-User Environment Tasks

In schools, offices, or online systems, many users may use the same computer system.

The OS manages this environment by:

- Creating separate user accounts
- Protecting user privacy
- Sharing Resources Fairly (Resources like CPU, memory, printers, storage)
- Preventing unauthorized access

## Creating and Managing User Accounts

The OS allows different users to have separate accounts with their own:

- files
- passwords
- settings
- desktop environment

**Example:** In a school computer lab, each student logs in using their own username and password. This keeps everyone's work separate and secure.

## Architecture of an Operating System

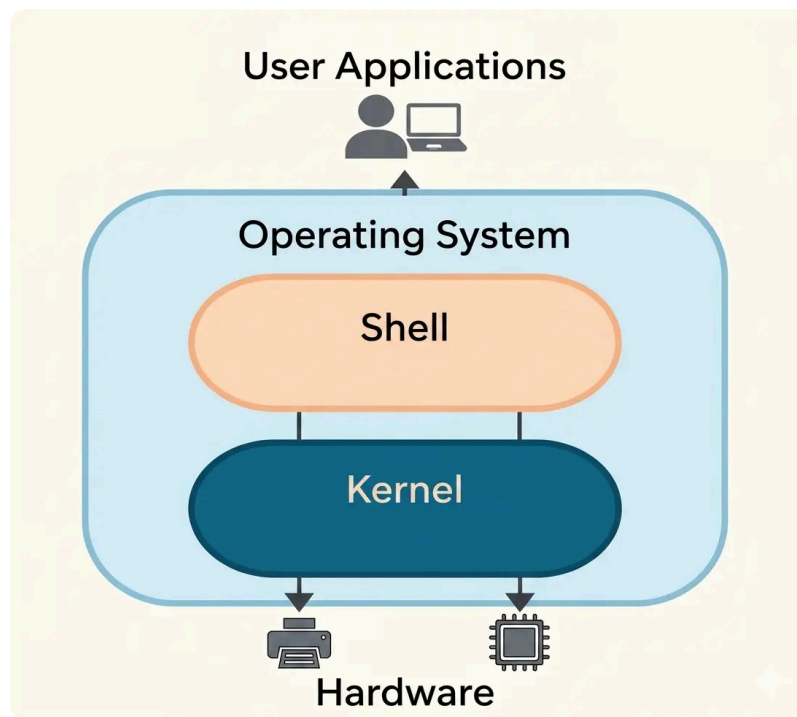
OS architecture means how different parts of the operating system are organized and work together. Each part performs a special task, and together they make the computer work properly.

**Example:** Just like a school has:

- administration
- teachers
- support staff

**All working together, the OS also has different parts working together.**

## Structure or Diagram



## Kernel vs Shell

**Kernel:** Core part of the OS that directly manages hardware (CPU, memory, devices, processes). It works in the background.

**Shell:** Interface between the user and the kernel. It takes user commands (CLI/GUI actions) and sends them to the kernel.

**Example:** When you type a command or click an icon, the shell takes that input and sends it to the kernel for processing.

**Simple way to remember:**

- Shell = takes your command
- Kernel = does the actual work

## Types of Shell:

There are **2** types of Shell.

- Graphical Shells
- Command Line Shells

## Command Line Shell (CLI Shell)

A text-based shell where users type commands to interact with the operating system. It is fast and widely used by advanced users.

Example: Bash, Windows Command Prompt

## Graphical Shell (GUI Shell)

A visual shell that uses icons, windows, and menus instead of text commands. It is easier for beginners to use.

Example: Windows Desktop Environment, macOS Finder

## OS Layers and Modular Design

### Lower Layer

This layer is closest to the hardware. It directly interacts with system hardware like CPU, memory, and devices. It includes the kernel and device drivers.

### Middle Layer

This layer acts as a bridge between hardware and user applications. It handles important services like memory management, process management, and system calls.

### Upper Layer

This is the user-facing layer. It includes the user interface (GUI/CLI), system applications, and user programs that allow users to interact with the computer.

**Each layer depends on the layer below it.**

### ✓ Benefits of Layered Design:

- easier to manage
- easier to repair
- easier to upgrade without changing the whole OS

### ✓ School Example:

- Support staff → lower layer
- Administration → middle layer
- Teachers/students → upper layer

### ✓ Example 1: Hospital System 🏥

- Lower layer → Support staff (cleaners, technicians, helpers) → Take care of machines and hospital environment
- Middle layer → Management (doctors, administration) → Control resources, treatment plans, and decisions
- Upper layer → Patients → Use services provided by hospital

### ✓ Example 2: Restaurant System 🍴

- Lower layer → Kitchen staff (cooks, helpers) → Prepare food (work behind the scenes)
- Middle layer → Manager → Takes orders and manages food delivery
- Upper layer → Customers → Place orders and receive food

### ✓ Example 3: Bank System 🏠

- Lower layer → Bank servers and back-office staff → Handle data processing and transactions
- Middle layer → Bank employees → Manage accounts and operations
- Upper layer → Customers → Use ATM, mobile app, or visit bank

## System Libraries

System libraries are a built-in collection of reusable functions used by applications to interact with the operating system.

When you click “Save” in an app:

- The app uses system library functions to request file saving
- Then the operating system handles it using system calls

## Device Drivers

Device drivers are special programs that help the operating system communicate with hardware devices.

### ✓ Examples:

- Printer drivers
- Keyboard drivers
- Graphics card drivers

Without drivers, the operating system cannot properly control hardware devices.

## Multiple Choice Questions (MCQs) on Operating System

### 1. What is an Operating System (OS)?

- a. Hardware device inside a computer
- b. System software that manages hardware and applications
- c. A type of computer virus
- d. A programming language

**Answer: b. System software that manages hardware and applications**

### 2. Which of the following is an example of an operating system?

- a. Microsoft Word
- b. Google Chrome