**1.2.1 Hardware Abstraction Layers**

* A hardware abstraction layer (HAL) is software that handles all of the communication between the hardware and the kernel.
* The kernel is the core of the operating system and has control over the entire computer.
* In some instances, the kernel still communicates with the hardware directly

**1.2.2 User Mode and Kernel Mode**

Absolutely, here's a concise technical explanation:

"Windows operating systems have two important areas: user mode and kernel mode. User mode is where applications run. It's like a controlled environment where apps can't mess up everything. If an app crashes here, it's not a big deal.

Kernel mode is the powerful area where the operating system works. It can talk directly to hardware. But if something goes wrong here, it's a serious problem, and the whole system can crash.

Drivers, which help the operating system communicate with devices, can run in both modes. In user mode, each app has its own space, like separate rooms. In kernel mode, everything shares the same space. If there's a mistake here, it can affect everything.

So, user mode is safe for apps, and kernel mode is where the big decisions happen. If there's trouble in user mode, it's manageable, but if something goes wrong in kernel mode, it's a major issue."

**1.2.3 Windows File Systems**

1. **ExFat:**

* This is a simple file system.
* FAT has limitations to the number of partitions, partition sizes, and file sizes that it can address, so it is not usually used for hard drives (HDs) or solid-state drives (SSDs) anymore

1. **Hierarchical File System Plus(HFS+):**

* This file system is used on MAC OS X computers
* it is not supported by Windows without special software, Windows is able to read data from HFS+ partitions.

1. **Extended File system (EXT):**

* This file system is used with Linux-based computers.
* Although it is not supported by Windows, Windows is able to read data from EXT partitions with special software.

1. **New Technology File System:**

* This is the most commonly used file system when installing Windows. All versions of Windows and Linux support NTFS.
* Mac-OS X computers can only read an NTFS partition. They are able to write to an NTFS partition after installing special drivers.

**1.2.4 Alternate Data Streams**In the NTFS system, files are made up of different parts, like a name or a time when it was created. The actual stuff inside the file, like text or pictures, is stored in a part called "$DATA." We can also attach something called an Alternate Data Stream (ADS) to a file. This is like adding extra information to the file. Sometimes, programs use ADS to add more details about the file.

Think of ADS like a secret compartment in a book. It's easy to hide things there. But this can also be used for bad things. Imagine if someone put a secret harmful message in that compartment, and you didn't even know it was there.

**1.2.5 Windows Boot Process**

**1.Bios(Basic I/O system) Firmware**

* The Bios firmware starts with initializer phase I.e POST(Power-On self test).(to check if all devices are communicating or not)
* The last instruction of Post is to find MBR(master boot record ) which has a small program that knows how to find & load the OS.
* The Bios then executes that program

**1.UEFI(Unified extensive firmware interface)**

* UEFI boots by loading .efi program files(instead of mbr) present In special disk partition called EFI System Partition (ESP).
* It has more visibility than BIOS in boot process
* It is more secure as it stores  boot code in the firmware.

1. Whether the firmware is BIOS or UEFI, after a valid Windows installation is located, the ****Bootmgr.exe**** file is run. ****Bootmgr.exe**** switches the system from real mode to protected mode so that all of the system memory can be used.
2. If the computer is coming out of hibernation, the boot process continues with****Winresume.exe.****
3. If the computer is being booted from a cold start, then the ****Winload.exe****file is loaded

**1.2.6 Windows Startup**

**\*\*Starting Programs Automatically:\*\***

Your computer can automatically start certain programs and services when it boots up. It can be controlled by 2 important folders:

1. **\*\*HKEY\_LOCAL\_MACHINE:\*\*** This is like a main folder in the computer's settings. It stores information about things that should start when the computer starts, like services that are needed for the system.

2. **\*\*HKEY\_CURRENT\_USER:\*\*** This is like a folder just for the person using the computer. It has info about things that start when that person logs in, like special services or programs they need.

**\*\*Different Types of Starting:\*\***

Inside these folders(above mentioned), there are different types of entries that tell the computer what to start. These include Run, RunOnce, RunServices, RunServicesOnce, and Userinit. These entries define which things should start and when.

**\*\*Using a Safe Tool:\*\***

You can manually add these entries to the settings, but it's safer to use a tool called Msconfig.exe. This tool lets you see and change what starts up when your computer does. You can find the Msconfig tool by searching for it.

**\*\*Msconfig tabs:\*\***

When you open the Msconfig tool, you'll see a window with five tabs:

1. **\*\*General:\*\*** Basic options for your computer's startup.

2. **\*\*Boot:\*\*** Settings related to booting up.

3. **\*\*Services:\*\*** Information about different services.

4. **\*\*Startup:\*\*** Programs that start when your computer does.

5. **\*\*Tools:\*\*** Additional tools and settings.

In a nutshell, you can control what programs and services start automatically when your computer boots up. The settings for this are stored in certain places in your computer's settings, and you can use the Msconfig tool to manage them easily.

**1.2.7 Windows Shutdown**

It's important to shut down your computer properly before turning it off. If you just switch it off, it can harm open files, services, and apps that are still running

****Ways to Shut Down:****

* ****Start Menu:**** Go to the Start menu and choose the power option.
* ****Command Line:**** You can also use a command called "shutdown" to turn it off.
* ****Ctrl+Alt+Delete:**** Press these keys and click the power icon to shut down.

****Different Shutdown Options:****

* ****Shutdown:**** Turns off the computer completely.
* ****Restart:**** Restarts the computer.
* ****Hibernate:**** Saves everything and lets you start where you left off when you turn it on again.

**1.2.8 Processes, Threads, and Services**

****Processes and Threads:****

* A Windows application is made up of processes.
* A **process** is any program that is currently executing(in windows application).
* Each process is made up of at least one thread.

A **thread** is a part of the process that can be executed(They are like workers who does the actual work).

****Task Manager:****

It's like a supervisor,which can be used to see and manage these tasks(processes).

Note: All of the threads dedicated to a process are contained within the same address space.This stops them from messing up each other's work.

* Windows can work on many tasks at once, depending on how many processors the computer has.

****Services:****

Some tasks are very helpful and work quietly in the background. These are called services.

You can change service settings by searching for "services."

**1.2.9 Memory Allocation and Handles**

* A computer works by storing instructions in RAM until the CPU processes them.
* The virtual address space is provided for a process that the process can use.
* Each process in a 32-bit Windows computer supports a virtual address space of 4(GB) gigabytes.
* Each process in a 64-bit Windows computer supports a virtual address space of 8(TB) terabytes.
* Each user space process runs in a private address space
* When the user space process needs to access kernel resources, it must use a process handle.
* A powerful tool for viewing memory allocation is RAMMap.

 RAMMap provides a wealth of information regarding how Windows has allocated system memory to the kernel, processes, drivers, and applications.

**1.2.10 The Windows Registry**

****Windows Registry - The Information Hub:****

* Registry is a big database that windows use to store important details about hardware, apps, users, and settings.
* It also tracks how these things work together, like what files apps use and how folders work
* Only those with special permissions (administrative privileges) can add, change, or delete keys and values in the registry. We use a tool called regedit.exe for this.

****Hierarchy and Hives:****

* The registry is a hierarchical database where the highest level is known as a hive.
* Inside each hive, there are "keys." These are like folders on the library shelves. Keys can also have smaller folders inside them called "subkeys."
* The values(It is something which contains information) are kept inside keys or subkeys.

**Hive -----> Keys -------> SubKeys --------> Values**

* In the registry, we can go up to 512 levels deep. It's like having lots of tiny folders inside bigger folders.(It is called ****Depth of Keys****)

****Main Hives - Key Players:**** Think of hives like big branches(folders at highest level). We have five main Hives:

* HKEY\_CURRENT\_USER (HKCU) : Holds information concerning the currently logged in user.
* HKEY\_USERS (HKU) : Holds information concerning all the user accounts on the host.
* HKEY\_CLASSES\_ROOT (HKCR): Holds information about object linking and embedding (OLE) registrations.
* HKEY\_LOCAL\_MACHINE (HKLM): Holds system-related information.
* HKEY\_CURRENT\_CONFIG (HKCC) : Holds information about the current hardware profile.

****Key Types and Values the hold:****

Keys in the registry can hold either more keys or values. Different values can hold different things:

* **REG\_BINARY**: Holds Numbers or true/false values
* **REG\_DWORD**: Numbers bigger than 32 bits or raw data
* **REG\_SZ**: Text values

**1.2.11 Lab - Exploring Processes, Threads, Handles, and Windows Registry**

**1.2.12 Check Your Understanding - Identify the Windows Registry Hive**