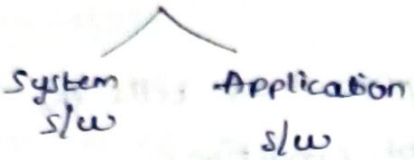
Software Engineering:

Application of systematic discipline and quantifiable approach to the development of operations and maintenance.

- s/w engineering paradigms
- computer software



System software:

- System s/w is a collection of software and hardware
- The system s/w required for controlling and managing and integrating the hardware components of the system.
- It is needed for the functioning of computer system.
- The system s/w is to be loaded in the computer system before using the system for performing any tasks. It usually interacts with hardware and applications
- It created a programming environment for application programs. The functions which are not available at h/w level are provided by the system s/w.
- the application program cannot be executed directly on the computer. Before execution of application program, system software is to be loaded onto the computer
- this acts as an interface b/w h/w of the computer system and application programs.
- All the information processing in the computer system are managed and supported by the system software.
- System software is the software which makes the computer system functional.

* The basic functions in computer system has stored memory management that can divided into file management, device management.

* Examples of system software:

OS, translators (compilers, interpreters),
device drivers, system utilities

OS:

It is a collection of system program that tells the computer what to do under variety of conditions. when the OS start the computer system.

- It occupies the first layer of software loaded in the computer memory.
- the system software provides a software platform and common are services needed by the other softwares. It performs resource management and provides interface b/w user and machine.

A resource may be the processor, memory or I/O devices, simply we can say that OS is responsible for these resource management.

Device Drivers:

- the computer system uses different devices such as keyboard, mouse, monitor, printer etc.,
- the functionality of these devices is controlled by the system software. the software varies from device to device called Device Drivers. of the device.
- the device driver is a system program which controls the operation and proper functioning of the device.
- Here, a device cannot be operated w/o its driver.
- A device driver is associated with the operating system.

- Each device has its own drivers and also its set of specialized commands
- when the computer system needs the use of device, the processor issues general command to the drivers of the device.
- The device drivers work as a translator b/w device and program.

Translators:

- The computer can understand only machine languages which consists of 0, 1s.
- writing programs using machine language is a very tedious & difficult job.
- High level programming languages are developed which is use english language.
  ex: C, C++, pascal, COBOL, Fortran
- The programs also written using the instructions available in the instruction set of processor
- when the program is to be executed on the computer it should be in machine language

System utilities:

Like H/w, software in a computer system also needs maintenance. The system utility programs are used for this, which supports enhance & secure existing programs and data in computer system.

They are small programs which performs specific tasks assigned to them.

The OS provides some utilities:

* file management
* Backup
* Data recovery
* virus protection

File management:

* The Data file management is made easier with this system utilities.
* many programs are written to help users to perform the following tasks - such as: searching files.

Back up:

In many organizations, the data is kept intact. this situation may exist which corrupt data files or delete them by accident. In such cases, data backups are used to obtain orginal data.

Data Recovery:

The data stored in computer system may get deleted of become inaccessible because of the following reasons -

1) System probs
2) viruses
3) electrical failures.

The loss of data can be recovered can using data recovery utilities.

Disk management:

(i) Defragmentation: In this process, fragments of files are kept sequentially on to the disk. It is easy to recognize the data stored on the disk. this reduces the time required to access the files.
The data gets arranged more effectively.

(ii) Formatting tools:

Formatting tools are used to formatting disk tracks and sectors allow to store data orderly.
The subdivision of track is called sector.

(iii) Data compression:

the formatting procedure generates the disk space.

(iv) Firewalls:

It acts as barrier b/w (system) computer.

this system utility required by the computer networks in an organization.

An unauthorized systems accessing information, it should be terminated by the firewall.

confidential data files, emails also can be protected with organization firewalls.

The firewalls provide protection to the computer system.

(v) Virus protection:

Viruses are small programs written with bad intention. They can damage the working of computer system. The security against the viruses can be provided by the antivirus programs.

The virus copy themselves to the disk from internet or other systems. the viruses are spread among the network computer connected together (or) storage devices such as floppy / CD Rom.

Application software:

The system software is necessary to run the computer system. It manages the computer system. But the computer system cannot be used for performing different tasks, just by loading system software.

So, it needs application software to accomplish a task. Different application softwares are needed to perform different tasks.

* Application software needs system software for their use. Every application software controlled by the system software.
* The Application s/w performs specific tasks.
* The application s/w is a s/w that the users will use.

ex: web applications

mobile applications

Desktop applications

* The system software can't notified by the users. It performs the functioning & managing of the computer system.
* The system s/w performs most of the background tasks. The users directly contact with application s/w.
* System s/w are programming softwares which are related to internal softwares.
* Daily used softwares are application softwares

ex: Banking systems word processors

Industries

Education systems.

Evolution Of Art to Engineering Discipline.

Art → craft → Engineering Discipline

program product	software product
1) developed by individual user	1) group of users
2) limited functionality	2) high functionality
3) debugging is difficult	3) debugging is easy
4) Documentation is good	4) Documentation is excellent.

* Software engineering principles have evolved over the last 60 years with the contribution of various researchers and software professionals.
* From the beginning period, of the software engineering acts as art. After that, with time it transformed into craft and finally to Engineering Discipline.
* Initially programmers used an Ad Hoc programming style. Ad hoc programming is an approachable solution in an unplanned manner. In this type of programming style, no plan is created on how to create structure and steps to complete the programming fast but w/o having any systematic approach.
  The problems need to solved in the required time.
* The Ad hoc program will cause various problems which results in less efficiency & another approach that is Systematic approach is adopted.

Software Development Life Cycle (SDLC):

Software process model -

It is a simplified description of a s/w process which is presented a particular perspective.

- The process models include which are part of the software process, s/w products and role of the people.

Process models:

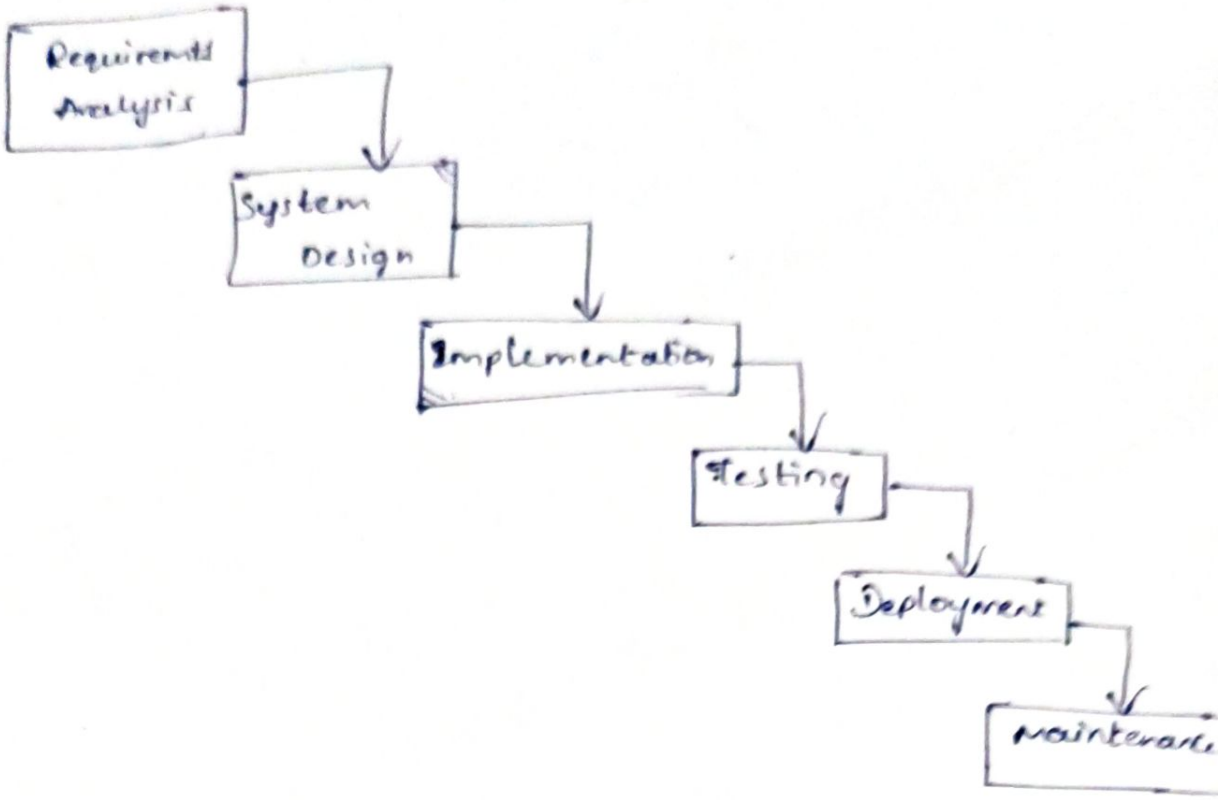
1) Linear sequential model (or) waterfall model
2) Prototyping model
3) RADI model (Rapid Application Development Model)
4) Evolutionary SPM
   - Iterative / Incremental model
   - Spiral model
5) Component based Development model
6) The Formal model

SDLC Phases:

BRS
↓
Business Requirements Specifications

1) Requirements & Gathering
2) Analysis & planning
3) Design
4) Implementation
5) Testing
6) Deployment & Integration
7) Maintenance

Waterfall model : (classical life cycle model)



Prototype model :