

# Software Engineering

Software is a program or set of programs containing instructions that provide desired functionality. And Engineering is the process of designing and building something that serves a particular purpose and finds a cost-effective solution to problems.

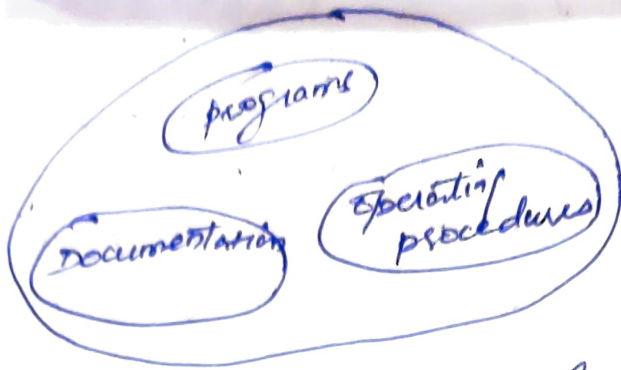
Software Engineering is a systematic, disciplined, quantifiable study and approach to the design, development, operations and maintenance of a software system.

According to Fritz Bauer defined Software Engineering as "The establishment and use of sound engineering principles in order to obtain economically developed software that is reliable and works effectively efficiently on real machines".

According to Stephen Schach defined "A discipline whose aim is the production of quality software, software that is delivered on time within budget, and that satisfies its requirement".

## Program Versus Software

Software is more than programs. It consists of programs, documentation of any facet of the program and the procedures used to setup and operate the software system.



## fig: Components of software

Any program is a subset of software only if documentation and operating procedure manuals are prepared.

programs is a combination of source code and object code. operating procedures consists of instructions to setup and use of the software system and instructions on how to react to system failure.

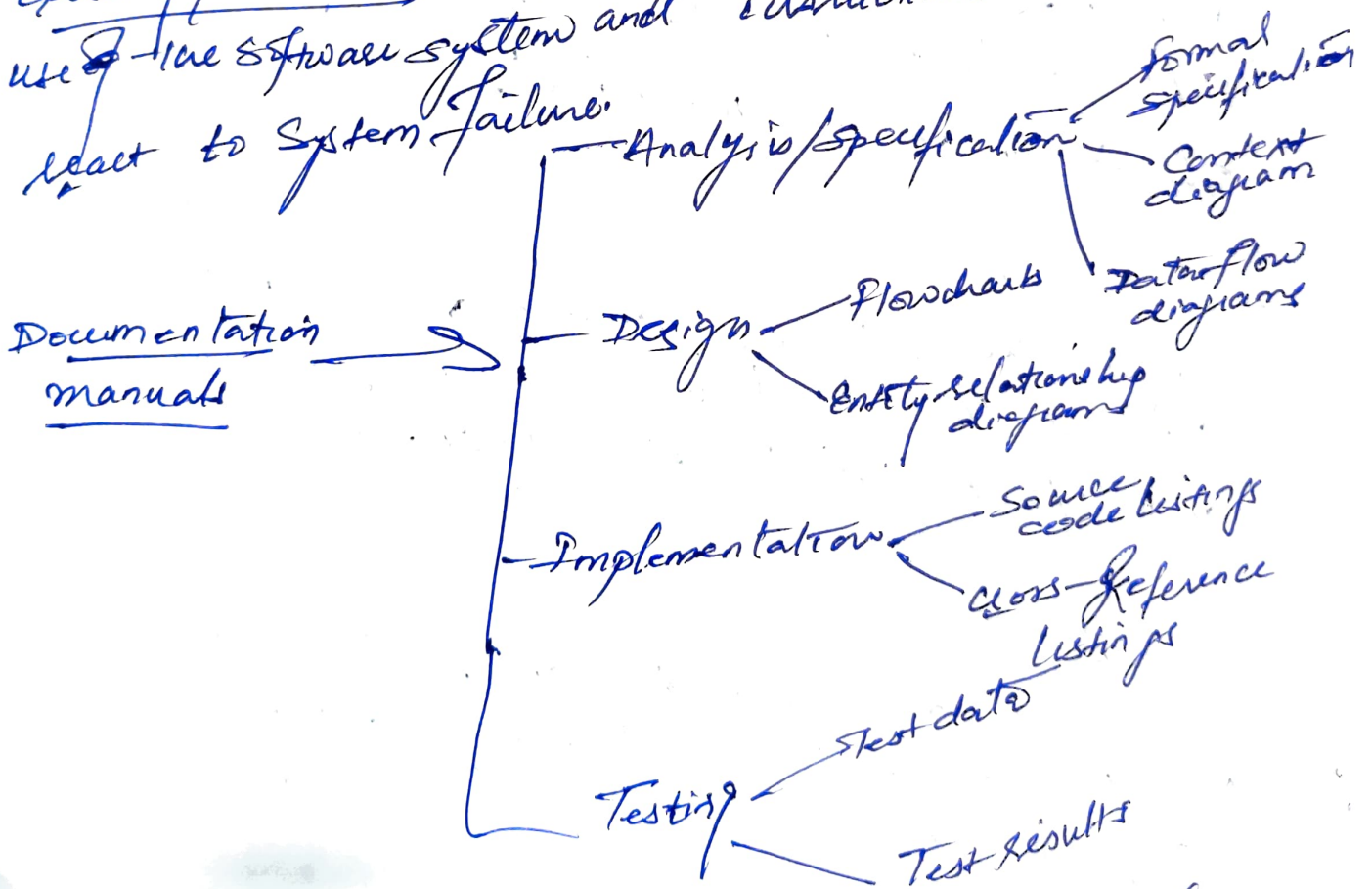


fig: list of documentation manuals

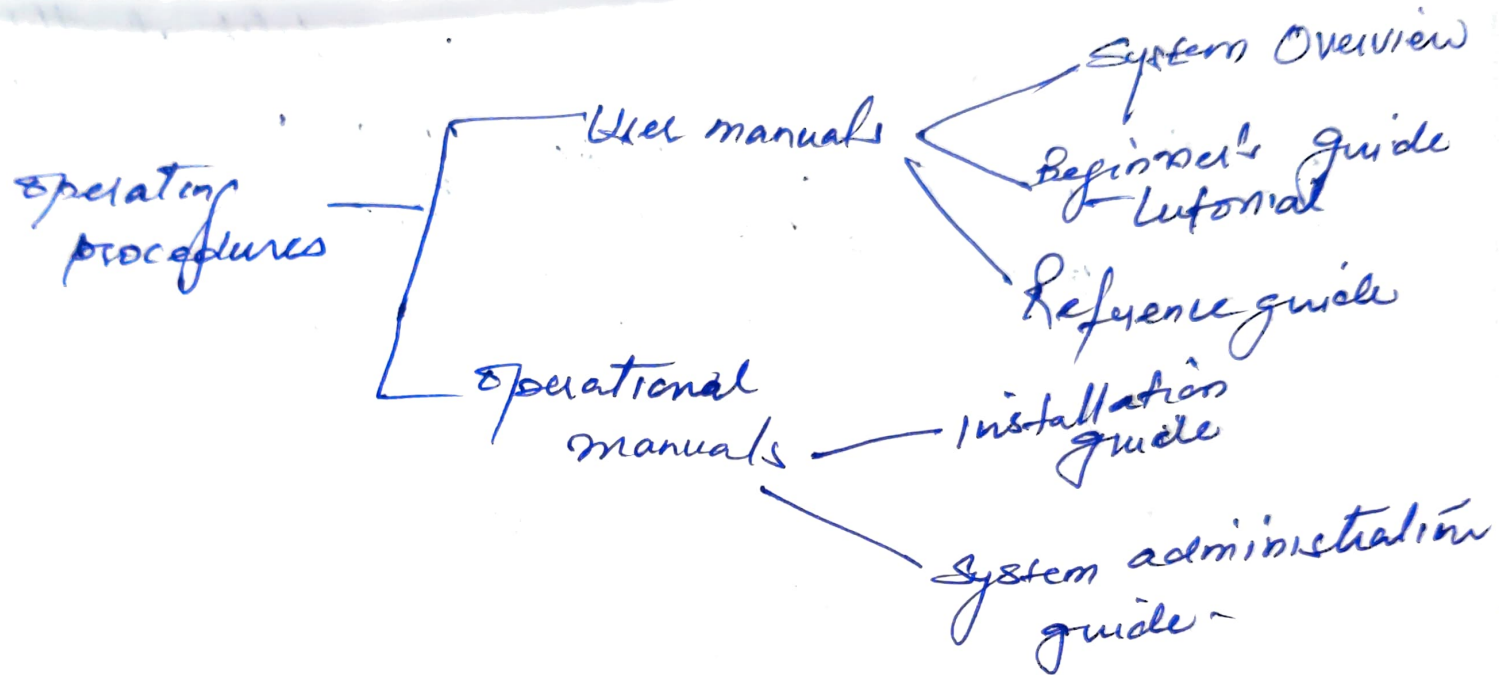


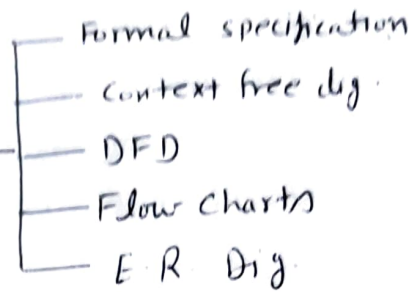
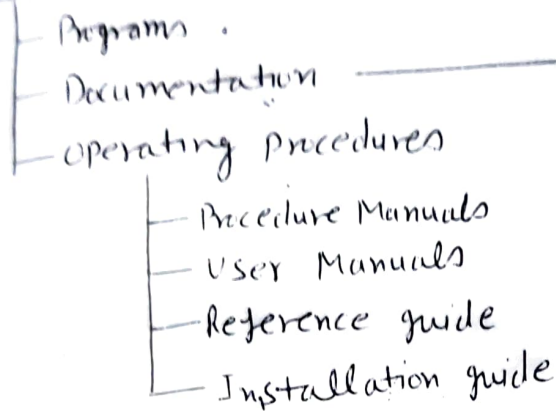
Fig: List of operating procedure manuals



## SLW Engg

→ SLW Engg has the objective of solving the problems of cost, timeliness, maintenance & quality by producing good quality, maintainable SLW on time, within budget

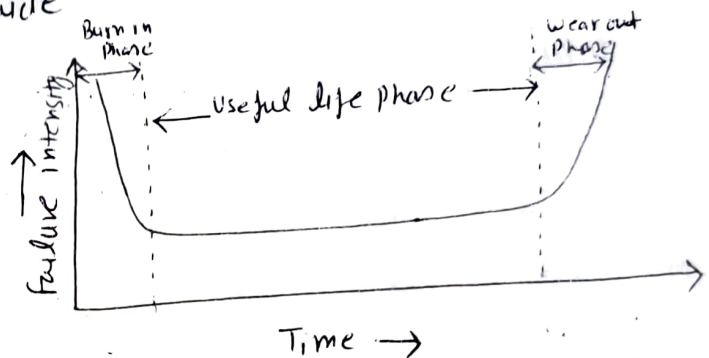
### Components of SLW:



### Characteristics of SLW:

1. SLW does not wear out →

\* There is no wear out phase in SLW life cycle.



Bath tub curve of SLW life cycle.

2. SLW is not manufactured →

\* making 1000 copies of a SLW is not an issue. & it does not involve any cost.

3. Reusability of Components →

\* In SLW every project is a new project

\* SLW Reusability - component based SLW engg.

4. SLW is flexible →

\* A Program can be developed to do almost anything.

## Categories of SW Application

1. System SW → collection of programs provided services to other  
eg Drivers, O.S.
2. Real Time SW → Monitor real world events  
as weather forecasting
3. Embedded SW → handles h/w components  
as ROM, automobile, security system, signalling system,  
control units of Power plants
4. Business SW → Process business Application  
as Data warehousing tool, Payroll, employee mgmt,  
account mgmt, ERP
5. Personal Computer SW → Used in personal computers.  
eg word processors, computer graphics, multimedia & animation,  
DBMS, computer games
6. Artificial Intelligence SW → uses non numerical algo to solve  
complex problems.  
eg are Expert systems, ANN, signal processing SW
7. web based SW → related to web applications.  
eg CGI, HTML, Java, Perl, DHTML
8. Engg & Scientific SW → scientific & engineering application SW  
CAD/CAM, MATLAB, Engg Pro, CKT analyzers.

## SDLC (Software Development Life cycle)

- (1) SDLC is a collection of processes that are followed to develop a software.
- (2) SDLC is a methodology that defines some processes which are followed to develop a high-quality software.
- (3) It covers the detailed plan for building, deploying and maintaining the software.
- (4) The main aim of SDLC is to define all the tasks required for developing and maintaining software.
- (5) It is followed for a software project within a software developing organization.

### Phases of SDLC

- 1) Requirement analysis
- 2) Feasibility study
- 3) Design
- 4) Coding
- 5) Testing
- 6) Deployment
- 7) Maintenance



## Phase 1: Requirement Analysis

- (1) It is the first phase of SDLC in which all the necessary information is collected from the customer to develop the software as per their expectation.
- (2) Some important questions like: what is the need of software, who will be the end user, what is the future scope of that software etc. are discussed.
- (3) The main aim of this phase is to collect the details of each requirement of the customer so that the developers will clearly understand what they are developing and how to fulfill the customer's requirements.

## Phase 2: Feasibility Study

- (1) It is the second phase of SDLC in which an organization discusses about the cost and benefits of the software.
- (2) It is an important phase because profit from the software plays an important role as if cost is very high then company may face loss.
- (3) After the feasibility study, the project may be accepted, accepted with modifications or rejected.
- (4) It measures how much beneficial the product is for the organization.

### Phase 3: Design

- (1) It is the third phase in which architects start working on logical designing of the software.
- (2) In this phase a SRS (System Requirements Specification) document is created which contains all logical details like how the software will look like, which language will be used, database design, modular designs etc.
- (3) This phase provides a prototype of the final project.
- (4) Basically, all it includes is design of everything which has to be coded.

### Phase 4: Coding

- (1) When the designing of the software is completed, then a group of developers start coding of the design using a programming language.
- (2) The interface of the software and all the internal workings according to the design phase is implemented in coding phase.
- (3) A number of developers code the modules and then all modules are arranged together to work efficiently.



(4) It is the longest phase of SDLC.

### Phase 5: Testing

- (1) Once the software development is completed, then it is sent to the testers. The testing team starts testing the functionality of the entire system.
- (2) In this phase, the software is checked for bugs or errors.
- (3) Whenever a bug is found, then the software is sent to the coders to fix it and then overall software is re-tested.
- (4) This is done to verify that entire application works according to the customer requirement.

\* Bug: Bug is nothing but formal name of error. When test engineers find any error in the application - they called it as bug. Bug is informal name to defects which is flaws or mistakes in the code or design.

Error is any mistake done by the user which hampers the application or software.

## Phase 6 : Deployment & Documentation.

- (1) After overall testing of the software and after checking that is bug free, then the software is launched and available for the users to use it.
- (2) Even after deployment of the software - if any bugs or errors are still found then the software is re-evaluated by the maintenance team and then it is re-deployed with a new version.

## Documentation

Documentation is the process of writing down every stage and each and every details of life cycle development so that anyone who follows this process may be able to do it in the real scene.



## Phase 7: Maintenance

- (1) The maintenance team look over the software usage and user's feedback.
- (2) Maintenance is necessary to eliminate errors in the system during the working life and to tune the software.
- (3) The bug fixing, upgrade and enhancement of the software is looked over by the maintenance team.