SE Unit 1

1

Introduction to s/w Engineering → This term is composed of two terms Software & Engineering.

S/w :- It is considered to be collection of executable codes, associated libraries and documentations.

Engineering :- Engineering is all about developing products, using well defined, scientific principles and methods.

So we can define SE as an engineering branch associated with the development of s/w product using well defined principles, methods and procedures. The outcome of SE is an efficient and reliable s/w product. when made for a specific requirment is called a s/w product

Need of S/w Engineering :-

1. Large software :- It breaks the problems of large projects into pieces.
2. Cost :- It can perform cost management by using various models.
3. Dynamic nature :- Due to dynamic nature of s/w there is always change in requirment for this SE is required.
4. Quality management :- Better process of s/w development provides better and quality s/w product. if number of user increases, it should be able to handle all the situations.

Characteristics of good S/w :- S/w characteristics can be defined on the following grounds.

① Operational :- This tells us how well S/w works in operations. It can be measured on the following basis:-
① Budget ② usability ③ Efficiency ④ Correctness ⑤ functionality
⑥ Security ⑦ safety.

② Transitional :- This is required when S/w is moved from one platform to another. It consist following basis.
① Portability ② reusability ③ Adaptability

③ Maintenance :- This aspect tells about how will a S/w has the capability to maintain itself in the ever changing environment. It consists:-
① maintainability ② flexibility ③ scalability.

S/w Components :- There are three basic components of S/w

① Program :- It is a collection of source code & object code written for performing for specific task. It can be a subset of S/w.

Apply in certain situation

(2) S/w document:- It consist of all the instructions pertaining to design, coding testing & maintenance. Good s/w consists of following documents.
(1) Analysis and specification (SRS)
(2) Design document (ER diagram, DFD etc)
(3) Coding document (Source code)
(4) Testing document (Test cases & results)

(3) Operating Procedures:- It provides information about the s/w i.e. How to install the s/w, How to control all the activities of the s/w. It is for helping operating staff & provide them the way to use and access it.
It can be divided into two parts.
(1) User manual:- step by step procedure of using s/w (User process)
(2) Operational manual:- It tells us the specifications of the device & technical characteristics. (Installation process)

Evolution of s/w Engineering:- The history of s/w Engineering begins in 1960s. It is as follows
1945 - 1965:- The origins (official start of term SE)
1965 - 1985:- The s/w crisis (Large no. of projects became fail during this period)

1990 – 1999 :- The internet Era.
2000 – 2005 :- The light weight methodologies (lighter version of S/w)
2010 – Till :- AI, ML, DS, Quantum computing etc.

S/w Crisis :- It can be defined as a set of problems raised during the s/w development periods. These problem can be occured due to rapid increase of computer power, advancement of technologies & complexity of the problem to be tackled.
These problems if remain unsolved results in incomplete s/w product.

Reasons of S/w Crisis:-

1. **The problem of scale :-** The methods used for developing small systems generally do not scale up with large systems.
2. **S/w is expensive :-** The cost of development of s/w is high as compared to the cost of h/w. Highly qualified and skilled man power is required to develop the s/w. So large amount can be paid which increases expenses.
3. **S/w is late :-** The term late means the s/w cannot be completed within the specified period of time. So it can't be delivered on time.
4. S/w is unreliable
5. Lack of
6. Code
7. Inconsi

Solution
1. Re
2. Des
3. P
4.
5.

(4) S/w is unreliable:- The unreliability means that s/w does not perform the required function. many failure occurs due to bugs that gets introduced into the s/w.

(5) Lack of understanding the problem
(6) Code was difficult to maintain
(7) Inconsistent productivity

Solution to s/w crisis :-

(1) Reduction in s/w budget
(2) Quality of s/w development
(3) Proper time scheduling should be done for completing the project on time
(4) Experienced team members.
(5) s/w must be deliverable i.e must be completed on time in all aspects

Similarity and differences b/w Conventional Engineering process and s/w Engineering process :-

Experiment/practical

Attention

Similarity:- ① Both require in-depth knowledge of their field.
② Both are trying to make world a better place
③ Both are getting automated slowly
④ Both includes some process for development i.e. design, manufacturing, production and plan etc.

Differences:-

	SE process	CE process
①	It is a process which majorly involves computer science, IT and discrete math	It is a process which majorly involves science math, and empirical knowledge
②	It includes computer and programming languages for building app/s/w.	It is related with building machines, hardware etc.
③	It emphasizes on quality, domain specific people, process identification etc.	It generally emphasizes on mass product domain specific component etc.
④	move from abstract design to abstract code.	moves from abstract design to concrete product.

S/w Quality attributes :- These are the features that facilitate the measurement of performance of a s/w product by s/w testing professionals. High score in SQA assures that a s/w application will perform according to the specifications provided by the client.

Various attributes :-

1. Safety :- It requires that systems to be build and configure in secure and reliable way.
2. Security :- mechanism to protect s/w against malicious attack.
3. Reliability :- s/w will not cause the failure of a system.
4. Resilience :- Ability of s/w to heal from unexpected event.
5. Robustness :- The degree to which a system or component can function correctly in the presence of invalid inputs.
6. Testability :- It refers to the degree to which any module can be verified as satisfactory or not.
7. Modularity :- Ability to divide the s/w into smaller modules.
8. Portability :- It refers to the usability of same s/w in different Environment.

(9) Usability :- It tells how easy user interface are to use.

(10) Reusability :- Reuse of any S/w in various formats.

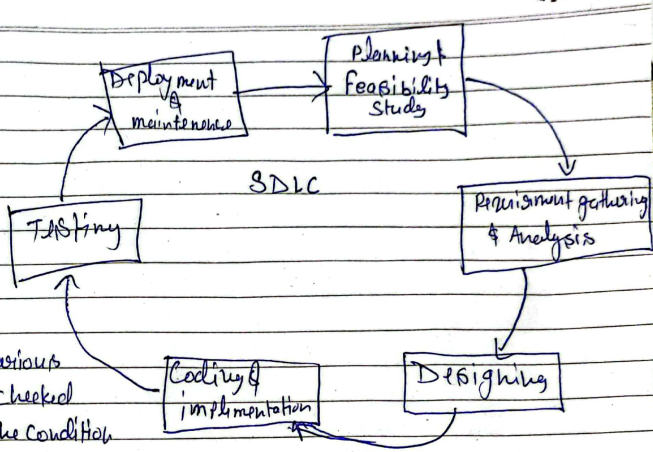
(11) Efficiency :- The ability to do something or produce something without wasting resources/ using less resources / by proper utilizing resources.

(12) Learnability :- Capability of S/w product to enable the user to learn how to use it.

S/w Engineering process :- It is a framework followed by organizations to produce high quality S/w product or S/w application. The complete process of developing the S/w is also known as S/w Engineering process

SDLC :- It includes all the phases/steps required by S/w developers from begining to end to complete the S/w ~~development~~ development

SDLC phases :-



① Planning & feasibility Study :-

This is the initial phase of SDLC. In this phase scopes and objectives are planned i.e. Selecting of the project, starting of the project are decided in this phase.
After this the feasibility on the various factors like budget, team etc are checked. Here we check that is we are in the condition of developing the SW or not?

② Requirment Gathering and Analysis :- This phase gather all the requirment and an them. It may consist of meeting stakeholders, interviews, work-shops for gathering information. After that requirment analysis is done for better under -standing of the requirment and then documenting the requirment in the form SRS document.

③ Design :- In this phase, the design specifications and the architecture of the products are created, documented and reviewed.

④ Coding & Implementation :- In this phase all the designed modules are implemented using some programming language.

⑤ Testing :- In this phase the QA team executes the test plan to evaluate the quality of the s/w. This phase consists finding and fixing bugs in the s/w.

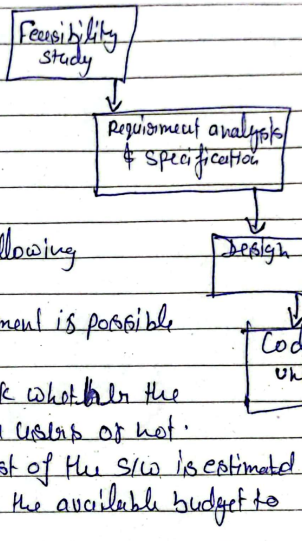
⑥ Deployment & maintenance :- After testing, if s/w is good then the application is deployed. The maintenance phase deals with maintaining and updating the s/w. After the delivery of the product if any error occurs, it is the responsibility of maintenance team to fix all the errors.

SDLC models :-

1. Waterfall model
2. Prototype model
3. Iterative Enhancement model
4. Spiral model
5. Evolutionary Development model

1970

① Waterfall model:- It is the very first SDLC model that is used for the development of S/W. It is also known as the linear sequential model, because in this model the outcome of one phase is the input for the next phase.
All other models are based on this model i.e this model plays the basis for all other models.



① Feasibility study:- The goal of this phase is to determine whether it is possible to develop the S/W or not. For this the available resources are analysed and checked if they are sufficient or not.
The feasibility can be measured in the following terms:-
① Technical feasibility:- Is the S/W development is possible with current technology.
② Operation feasibility:- It is used to check whether the developed S/W will be useful for the users or not.
③ Economic feasibility:- Under this the cost of the S/W is estimated and then compared with the available budget to check whether it is sufficient or not.
④ Schedule feasibility:- This analysed to estimate the completion time of S/W.

(2) Requirement analysis and specification :- This phase identify the customer requirement and document them properly. Two different activities are performed in this phase.

(i) Requirement Gathering and Analysis :- All the requirement are gathered from the customer and then analyzed to check if it is complete or not. Inconsistencies are also removed to identify actual requirement.

(ii) Requirement Specification :- In this, user requirements are systematically organized into a SRS document.

(3) Design :- The goal of this phase is to transform the requirements specified in SRS document into some structure that is suitable for implementation in some programming language i.e. S/W architecture is derived from SRS.

(4) Coding & Unit testing :- The purpose of Coding phase is to translate the S/W design into source code. Each component of design is implemented as a program module. During Unit testing phase each module is individually tested to determine the correct working of modules.

(5) Integration & System testing :- After unit testing all the modules are integrated in a planned manner. All the modules can not be integrated in single step rather it is carried incrementally over a no. of steps.

System testing is carried out when all the modules have been successfully integrated and tested i.e. After integrating all the modules a complete system testing is performed for checking its completness.

(c) Maintenance :- It involves performing any one or more of the following 3 kind of activities:

a :- Corrective Maintenance :- Correcting errors that were not discovered during the product development phase.

b :- Perfective Maintenance :- Improving the implementation of the system and Enhancing the functionalities of the system according to the customer requirement

c :- Adaptive Maintenance :- Porting the s/w in a new environment. i.e. changing the computer platform from windows to linux etc.

Advantages of waterfall model :- (1) Simple and easy to understand
(2) Suited for smaller projects where requirments are well defind

Criteria to use waterfall models :-
(1) Requirment is not changing rapidly (2) App. is not complicated and big
(3) Project is short (4) Requirments are clear in advance
(5) Technology and tools used are not dynamic.
(6) Resources are available.

Drawbacks of waterfall model:-

(i) No Feedback Path:- This model assume that no error will ever committed b
developers in any phase. Thus it does not include any mechan
for error correction.

(ii) Difficulty Accommodating Customer requirements:- It assume that all the
requirement has been defined at the beginning of the project
It does not provide path for any change ~~requirement~~ request after th
requirement specification phase is complete.

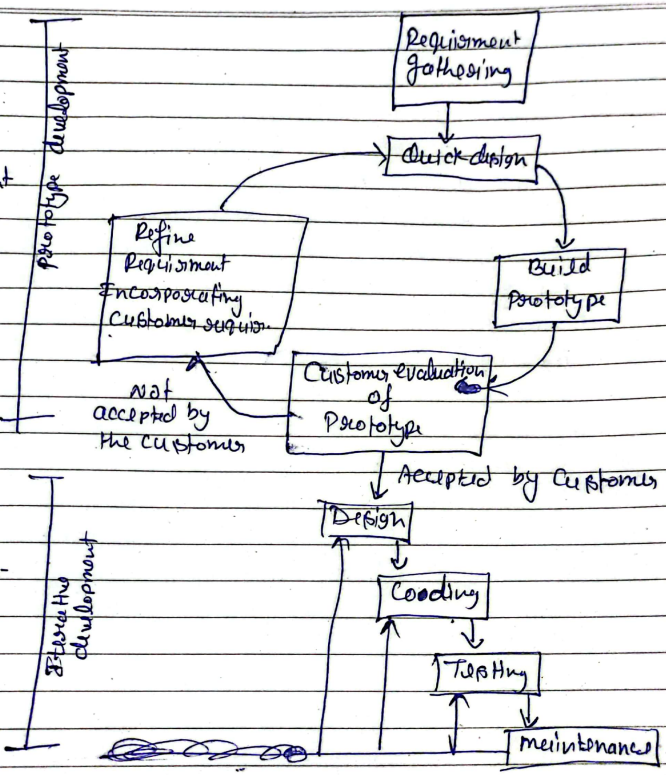
(iii) Absence of overlapping Phases:- In this new phase can be started only
after the completion of previous phase. Bu
to increase efficiency and reduce the cost phases needs to overlap.

(iv) Risk handling not supported:- This model has no mechanism fo
risk handling.

(v) Incremental delivery is not supported.

Prototype model:-

In this model a prototype of the actual S/w is build before the actual development of the S/w.
A prototype is a sample implementation of the system By using this prototype the client can get an actual feel of the system and can suggest changes, if needed.
Based on the customer feedback, the requirment are modified and accordingly prototype is also refined. This process continue till the customer approves the prototype.



Phases of Prototype :-

① **Requirment Gathering :-** Requirments are gathered from the customer for starting the development process.

② **Quick design :-** • After gathering requirment a rough idea about the product is identified and designing of the product is completed on temporary basis. So that further improvements can be done.
• A rough architecture is drawn based on that the prototype can be build.

③ **Build Prototype :-** Based on the architecture a prototype of the product is build. This prototype consist all the features defined by the customer.

④ **Customer Evaluation of Prototype :-** After building the prototype, it is handovered to the customer for evaluation i.e. if this prototype is meeting all the requirments defined by the customer or not.
if customer accepts and agree upon the prototype then actual designing phase is carried out followed by all the development phases i.e. coding, testing and maintenance.

⑤ Refine Requirements and incorporating Customer Suggestions :-

if customer suggests any changes after the evaluation of prototype, this phase carried out. In this the initial requirements are refined and then this cycle continues till the acceptance of the customer.

Note :- Design, coding, testing and maintenance all these phases are similar to waterfall model except that there will be return back path to previous stage.

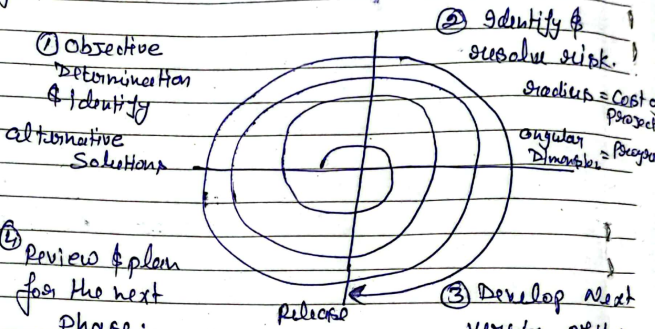
Advantages :- ① Errors can be detected much earlier.
② Quicker user feedback is available.
③ Users are actively involved in the development.
④ This model is flexible. Modification can be done easily.
⑤ Provides more significant customer satisfaction.

Disadvantages :- ① Time consuming → due to various changes in prototype
② High cost
③ Increases complexity

When to use Prototype model:-

1. When desired system requires lot of interaction with users.
2. Requirements are changing rapidly.
3. Where Risk is major factor.

Spiral model (1988):-



- If we want to analyze the risk at initial level of any S/W development, Then this model is best solution.
- This model looks like a spiral consisting of many loops. Each loop of the spiral represents the phase of S/W development process.
- The exact number of loop is not fix it can vary from project to project as per the requirment.

- The radius (r_1, r_2, r_3, r_4) of the spiral represents the cost i.e. the radius $\propto$ Cost. if radius increases then cost would also increase.
- The angular dimension represents the progress in the current phase.

Each phase of the model is divided into four quadrants. The functions of each quad. as follows:-

① Objective determination and identify alternative Solutions :- After gathering the requirements from the customers. objectives are identified and analyzed. After analyzing the requirements of the customers, the alternative solutions are proposed in this quadrant.

② Identify and resolve risk.

- In this quadrant, after evaluating all the possible solutions, the best possible solution is selected. After this the risks associated with that solution are identified.
- After identifying risks, they are resolved using best possible solution.
- Best way to select the possible solution is to build prototype i.e. by creating the prototype we can check and identify the best possible solution and also we can identify risk.

③ Develp next version of the product :- After identifying risk in prototype, the next version is created by incorporating all the changes to resolve risk.

In this quadrant all the S/W development steps i.e. design, coding, testing, implementation are performed to complete the S/W.

④ Review and plan for the next phase:- In this quadrant the developed version of the S/W is developed to the customer for evaluation. If customer satisfies, the development stops here, & if the customer is not satisfied, then planning for the next phase is done by the development teams for the appropriate development of the S/W.

Criteria to use Spiral Model:-

① When project is large & Complex.
② When risk and cost evaluation is required initially.
③ When requirements are unclear and changing.
④ Can be used to medium to high risk projects.

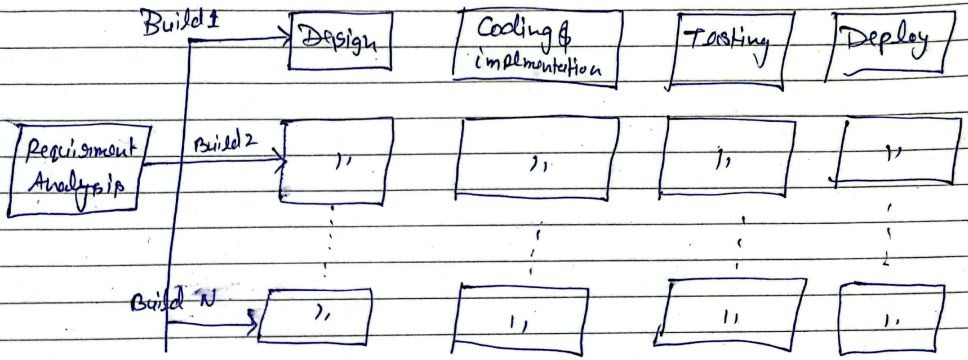
Advantages:-

① Risk analysis is performed earlier.
② Cost-estimation is easy due to fragmented development.
③ Customer interaction is high, which satisfies clients easily.
④ Suitable for large and extensive projects.
⑤ Quality result.

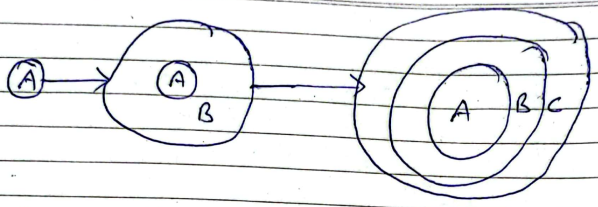
Disadvantages :-

1. Expensive and time taking.
2. Difficult in managing time.
3. Need well experts
4. Not suitable for smaller projects
5. Maintenance of this model is not so easy.

Iterative Enhancement model / evolutionary model :-



In this model the s/w is broken down into different modules which are incrementally develop and delivered to the users.



- Here A, B, C are the modules of S/W that are successively developed and delivered to the users, until the desired system is released.
- Each module follows the waterfall development phases i.e. requirement analysis, design, coding, testing & deployment.
- The development team first develops the core features of the system. Once the core feature i.e. module A is developed then next modules are developed by adding features to core module.
- At each successive versions S/W is constructed and delivered to the customer and feedback is to be taken. These feedbacks are incorporated in next versions.