19/7/19

UNIT - I

→ **SDLC** :-

Software Development Life Cycle.

→ **Program** :-

Collection of instructions which are executed sequentially or step by step.

→ **O.S.** :-

A sofware which controls all over the system to maintain input & output of the data.

→ **System** :-

It is a group of different-diff. components which work together to perform a particular task.

Scanned by CamScanner

★ Software Characterstics :-

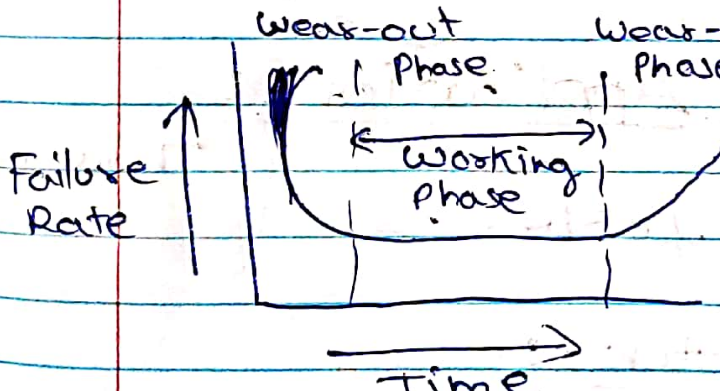
Software is a logical entity rather than a physical entity, therefore, software characteristics are quite different from hardware

1.) Software is Engineered or developed and not manufactured

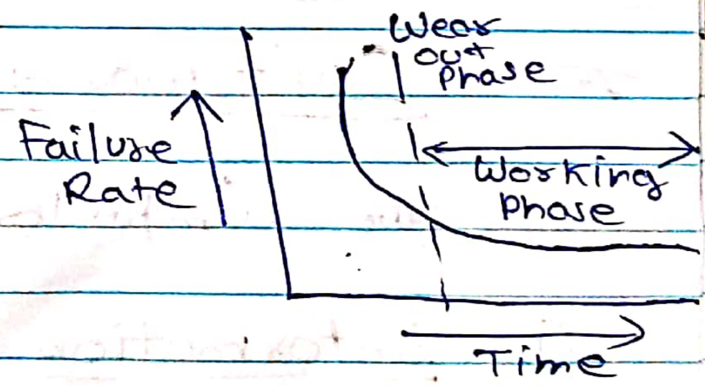
→ To engineer a software product, high quality is to be achieved through good design after critical analysis by peoples:

→ It requires application of correct methodology, model, and suffasticated tools to engineer a product.

2.) Software does not wear-out:-



H/W



S/W

→ Software does not wear out due to age, whereas in case of hardware it has relatively high failure rate, early in it's life, which accounts for designing or manufacturing defects.

→ Later these defects are corrected and the failure rate drops for some period of time.

20/7/19

3.) **Software is Custom-built:-**

Software products most often are developed in view of the customized customer requirements and system tend to customer specific.

→ Custom-Built software products under-take a lot of effort and time of the development team.

→ Reuse of components from the libraries help in reduction of effort.

★ **Software Applications:-**

There are two important factors:-

① **Information Contents:-**

They refers to the meaning and form of incoming and outgoing information.

to the predictability of the order and timing of information.

-> Some applications are determinant & some are indeterminant.

-> Determinant applications accept data that have a pre-defined order, execute analysis algorithm without interruption & produce resultant data in reports, whereas indeterminant app^ns accept input that have varied contents, execute algorithm that can be interrupted and produce output that varies with environment and time.

Categories of S/W App:

1) System Software:- It is collection of programs built to serve other programs.

Eg:- DOS, Windows, OS, etc.

2.) Real-Time S/W:- S/W that work on real world events where the response time is very crucial.

Eg- Data gathering Components, Monitoring Components etc.

3.) Application S/W :-
It is known as third party software which is to be installed on a device.

Eg - M.S. Office, etc.

4.) Engineering & Scientific S/W :-

CAD (Computer Added Design).

5.) Embedded S/W :-
It is used to make intelligent products.

Eg :- Microwave, etc.

6.) Personal Computer (P.C.) S/W :-

Eg - Word, Excel. which remains in a single device, if needed in another device, we have to install it again.

7.) Web Based S/W :-
S/w which are used for browsing ~~the~~ web.

Eg :- Chrome, etc.

8.) Artifical Intelligence (A.I.) S/w :-

Eg - Robot.

Scanned by CamScanner

20/7/19

★ **Software Engineering** :- It is a profession & field of study, dedicated to ~~design~~ designing, implimenting and modifying s/w, so that it is of higher quality, more affordable, maintainable and faster to bit.

→ S.E:- focus on

i) Improving the productivity of the development process.

ii) Controlling & predicting the cost of software development.

iii) Producing what the customers want.

iv) Producing s/w that ~~help~~ have features like Realibility, clearity, extensibility.

→ **Black Box**

Input → [Process] → Output

Don't care about Process

→ **White Box** :-

Input → Sees all Process → Output

22/7/19

~~A~~ Software Process Models :-

⇒ SDLC Steps :-

1.) Preliminary Investigation / Recognition of Need

2.) Feasibility Study

a) Technical F.S.

b) Economical F.S.

c) Operational F.S.

d) Legal F.S.

3.) Analysis → Detailed Study

→ Collection of Data

Data Gathering Tools —

a) Feedback

b) Interview

c) Questionnaire → ① Objective ② Descriptive

d) Onsite Observation - Data Flow

e) Reports, Manuals

4.) Design →

a) Output Design ←

↓

b) Input Design

↓

c) Process Design

↓

d) Storage Design

↓

e) Design Submitted to Top management

Accepted

No →

23/7/19

5.) Coding :-

Code written in selected language

6.) Testing :-

a) Unit Testing
b) System / Integration Testing

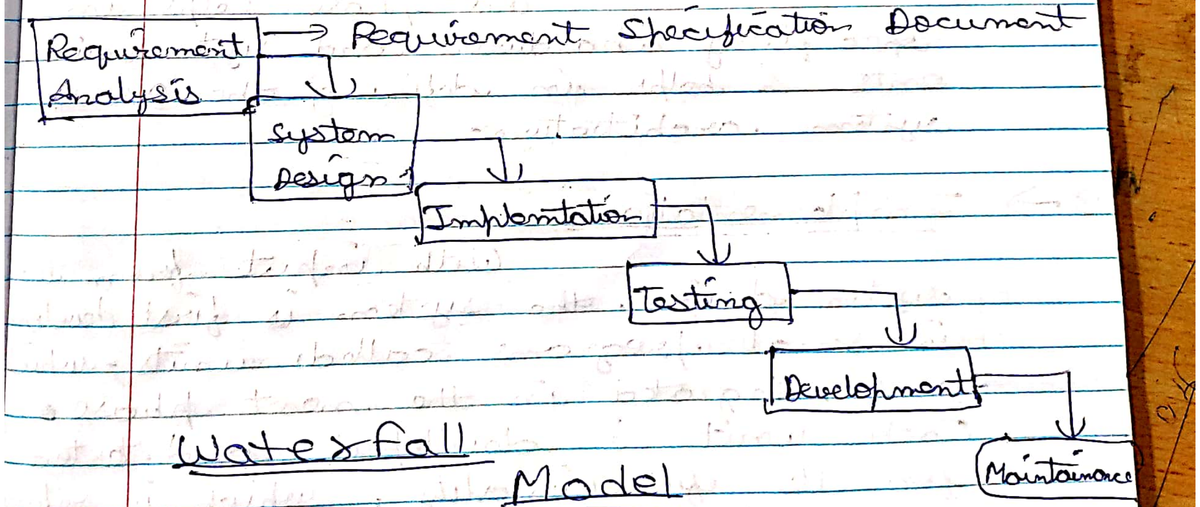
7.) Implementation :-

a) Parallel
b) Full

8.) Maintainance

9.) Post Implimentation

☆ Water Fall Model :-



Waterfall Model

→ It is also referred as a linear sequential life-cycle model, it is very simple to understand and use.

→ In this model, each phase must be completed before the next phase can begin and there is no overlapping in the phases.

1→ **Requirements gathering & Analysis** :- All the possible requirements of the system to be developed are captured in this phase & documented in a requirement specification document. No new requirements accepted after completition.

2→ **System Design** :- This phase helps in specifing hardware and system requirements & helps in defining the overall system architecture.

3→ **Implementation** :- With inputs from the system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality, which is referred to as unit testing.

23/7/19

4→ **Integration and Testing:-** All the units developed in the implimentation phase are integrated into a system after testing of each unit.

- Post integration, the entire system is tested for any faults and failures.

5→ **Deployment:-** After testing the product is deployed in the customer environment.

6→ **Maintainance:-** There are some issues which come up in the client environment, to fix those issues, patches are released.

- Maintainance is done to deliver these changes in the customer environment.

→ **Advantages:-**

i) Easy to manage due to the regidity of the model.
ii) Works well for smaller projects where requirements are very well understood.
iii) Process & results are well documented

23/7/19

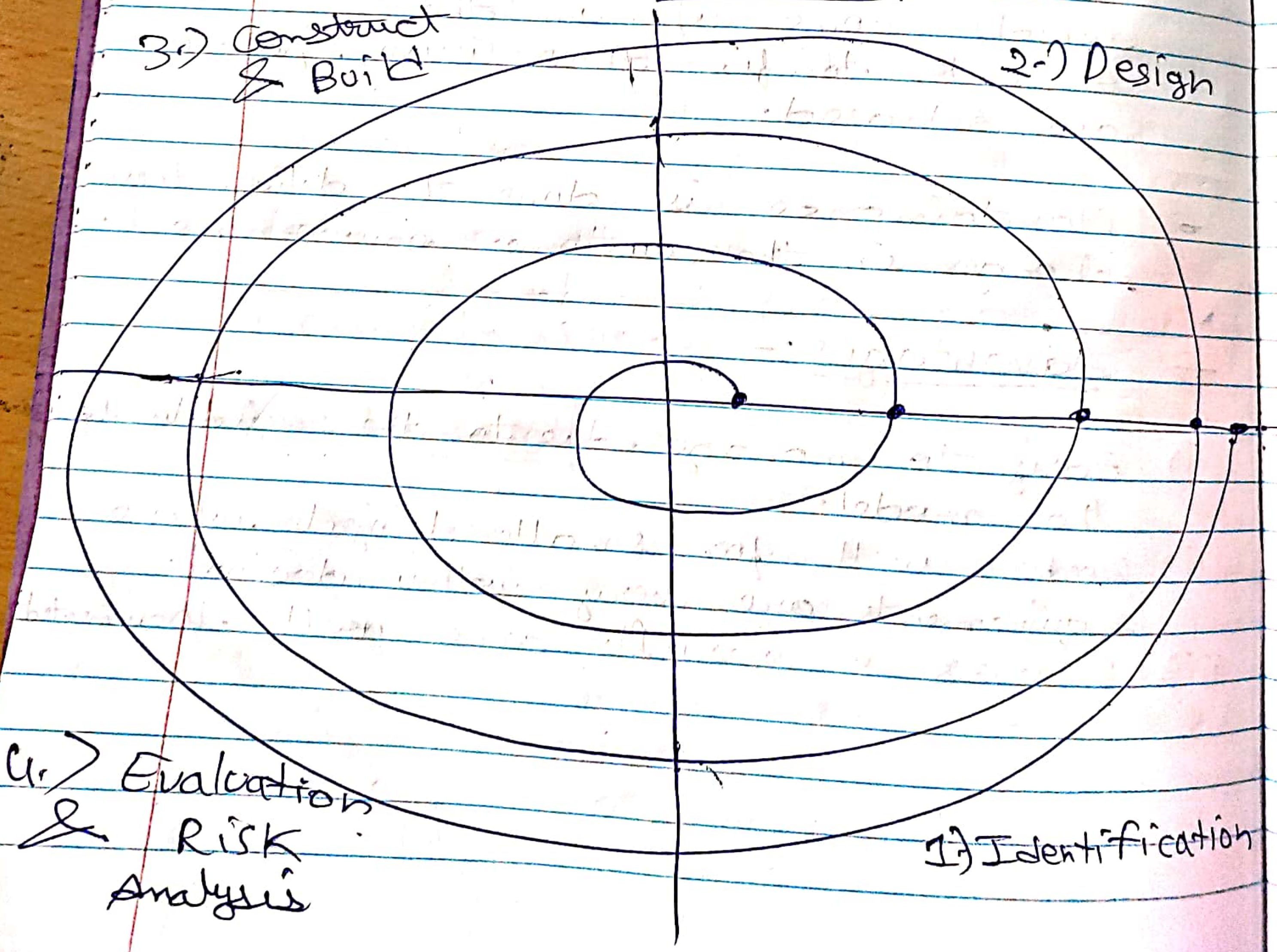
→ **Disadvantages :-**

i) High amount of risk & uncertainity.

ii) It is not a good model for complex and object oriented projects.

iii) Can not accomodate changing requirements.

☆ Spiral Model :-



It allows incremental release of the product for incremental refinement through which iteration around the spiral.

→ The spiral model has 4 phases:-

i). **Identification :-**

This phase starts with gathering the requirements.

→ In the subsequent spirals as the product matures, identification of system requirements, sub-system requirements and unit requirements are all done in this phase.

ii) **Design :-**

The design phase starts with the conceptual design in the base line spiral and involve architetal, logical design of modules, physical product design and the final design in the subsequent spirals.

iii) **Construct or Build :-**

The construct phase refers to production of the actual S/W product at every spiral.

In the subsequent spirals with higher quality on requirements and design details, a working model of s/w is produced with a version number.

→ These Builds are sent to the customer for feedback.

iv) **Evaluation & Risk Analysis:-**

Risk analysis includes identifying, estimating and monitoring the technical feasibility and management risk such as scheduling and ~~post~~ cost overrun.

→ After testing the build, at the end of first iteration, the customer evaluate the software & provides feedback.

→ Based on the customer evaluation, the s/w development process enters in the next iteration.

⇒ **Advantage:-**

i) Changing requirements can be accomodated.

ii) Allows extensive use of prototypes.

iii) Requirements can be captured more accurately.

User see the system early.

iv) User see the system early.

v) Development can be divided into smaller parts and the risk part can be developed earlier which help in better risk management.

⇒ **Disadvantage** :-

i) Management is more complex.

ii) End of the project may not be known early.

iii) Process is complex.

iv) Large number of intermediate stages require excessive documentation.

v) Not suitable for small or low risk projects and could be expensive for small projects.

29/7/19

~~A~~ Prototype Model :-

~~In~~ Prototype is a working model of a software with some limited function. The S/w prototyping refers to the building S/w App^{n} prototypes which display the functionality of the product under development But may not actually hold the exact logic of the original S/w.

⇒ These are 6 Phases :-

i) Requirements Gathering and Analysis :-

In this phase the requirements of the system are define in detail. During the process, the users of the system are interview to know what is their expectations from the system. This step involve understanding the very basic product requirements in terms of user interface.

ii) Quick Design :-

In this phase a simple design of the system is created. However, it is not a complete design. It gives a brief idea of the system to the user.

The quick design helps in developing a prototype.

iii) Build a prototype:- In this phase an actual prototype is design based on the information gather from quick design, it is a small working model of the required system.

iv) Initial User Evaluation:- In this stage the proposed system is presented to the client for an initial evaluation.

→ It helps to find out the strength and weakness of the working model.

→ Comments & suggestions are collected from the customer and provide to the developer.

v) Refining Prototype:- If the user is not happy with the current prototype, we need to refine the prototype according to the user's feedback and suggestions.

→ Once the user is satisfied with the developed prototype, a final system is developed based on the approved final prototype.

vi) Implement product & Maintain:- Once the final system is developed on the final prototype, it is tested and deployed to production.

→ The system undergoes a routine maintainance for minimizing and preventing failures.

⇒ Advantages :-

i) Users are actively involved in development. Therefore, errors can be detected in the intial stage of development process.

ii) Missing functions can be identified.

iii) Customer satisfaction exist because customer can feel the product at a very early stage.

⇒ Disadvantages :-

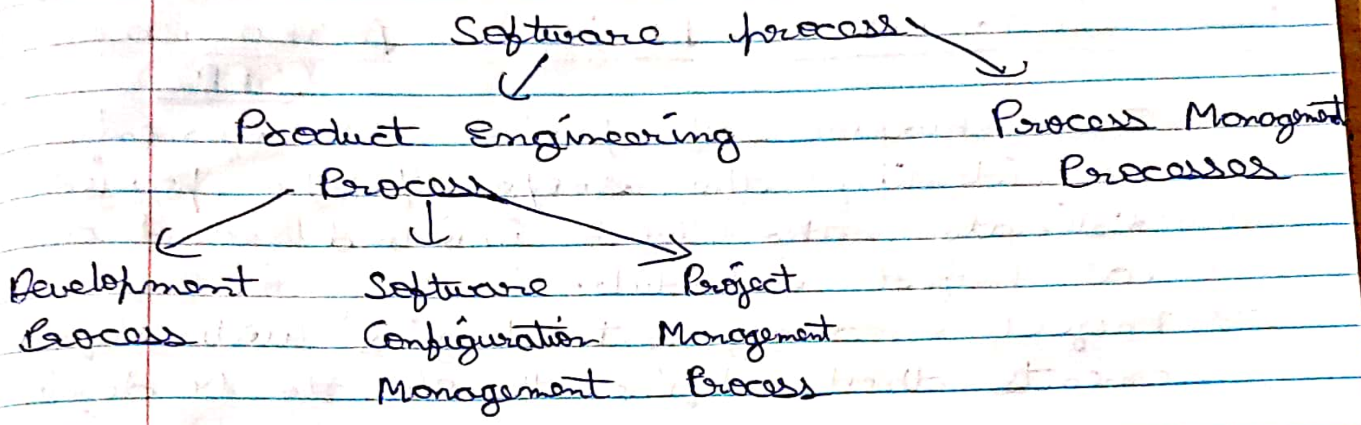
i) Prototyping is a slow & time taking process.

ii) The cost of developing a prototype is total waste.

iii) It is very difficult for S/W developers to accomodate all the changes demanded by the clients.

30/7/19

☆ Software Components :-


Software process
Product Engineering Process
Process Management Processes
Development Process
Software Configuration Management
Project Management Process

→ A component is a subsystem.

→ Every software program and application is a system and have different components.

→ The components of a software system can be its different section, forms & screens.

→ From the perspective of a S/W Architect, this system has main components such as frontend and backend services, libraries and classes.

- **Frontend** :- Interface for the user.

- **Backend** :- Process of storing data.

Concept of Project Management: /Planning

Software Project Management (SPM)

The purpose of project management is to identify the scope of the project, estimate the work involved and create a project schedule.
Project management begins with requirements that define the S/W to be developed.

The purpose of the project monitoring and control is to keep the team and management up to date on the project progress.

Fundamentals of SPM :-

Project management activities can be viewed as having their measure phases project planning, project monitoring and control and project termination.

The SPM process begins with a set of activities that are collectively called project planning.

During planning all the activities that management needs to perform are plant while during project control plan is executed and updated. Without a proper plan, no real monitory control of project is possible.

6:) The basic goal of planning is to look into the future, identify the activities that need to done, to complete the project successfully and planned the scheduling and resource allocation for these activities.

7:) Economic, political and personal factors should be taken into account for a realistic plan.

8:) The input to the planning activities is the requirements specification.

1.) **Project Scope** :-

2.) **Project Schedule** :- Every project has the particular time interval within which the project must be completed.

3.) **Project Team Organisation** :- Every project has an organised way under a project leader having a team to complete the project.

4.) **~~D~~Technical Description of the proposed System** :- Every proposed project has the strong technical description, so that, project can be easily implimented.

5.) **Project Standards, Procedures & proposed technical Tools** :-

6.) **Quality Assurance Plan**

7.) **Special Development Tools & Techniques**

8.) **Configuration Management Plan** :- Details of the product like RAM, ROM, etc.

9.) Documentation Plan :- Every project must have a planned documentation for the maintainance of the project of the latter stage.

10.) Data Management Plan :-

11.) Resource Management (Mgt.) Plan :- Resource which are availalable or to be purchased.

12.) Test Plan :- (Unit test / system test)

13.) Training Plan

14.) Security Plan

15.) Risk Management Plan :- All the limitation constraints and possibilities must be considered to avoid any type of risk.

16.) Maintainance Plan

13/8/19

⇒ Steps in Project Planning:-

→ The project planning is consider to be a process of estimating, scheduling and assigning the project resources in order to deliver an end product of best quality.

1.) Project Scope Definition & Scope Planning:-

In this step, we document the project work that would help us achieve the project goal.

→ We document the assumptions, constraints, user expectations, business requirements, technical requirements, project objective, project deliverables and everything that define the final product requirements.

2.) Quality Planning:-

The various factors influencing the quality of the final product are determined.

3.) Project Activity Definition & Activity Sequencing:-

The project activity sequencing identifies the interdependence of all the activities defined.

4.) Time, Effort & Resource Estimation:

The effort can be calculated using one of the many techniques available such as function-points, lines of codes, complexity of code, etc.

5.) Risk Factor Identification:- Expecting the unexpected and facing it. It is important to identify and document the risk factor with the project based on the assumptions, constraints, user expectations, specific requirements, etc.

6.) Schedule Development:- The time schedule for the project can be arrived at based on the activities, interdependence and effort required for each of them.

7.) Cost Estimation & Budgeting:-

8.) Organisational & Resource Planning:-

9.) Risk Management Planning

10.) Project Plan Development & Execution The project plan documents all the assumptions, activities, schedules & time line of the project

11.) Performance Reporting :-

12.) Planning Change Management :-

The req for changes need to be analysed carefully and its impact on the project should be studied.

13.) Project Rollout Planning :-

☆ Fourth Generation Techniques

5/8/19

Role of Metrics & Measurements

Software Measurements :- A measurement is a display of the size, quantity, amount or dimensions of a particular attributes of a product or process.

→ To access the quality of the engineered product or system and to better understand the models that are created, some measures are used. These measures are collected throughout the SDLC and with an intension to improve the software process on a continuous basis.

→ Measurements helps in estimation, quality control, productivity ~~asses~~ assessment and project control through out the software project.

→ There are 2 types of measurements:-

i) Direct Measures:- It include S/W proce like ~~fo~~ cost and effort applied an products like lines of code produced, execution speed and other defects that have been reported.

ii) **Indirect Measures:-** It includes product like functionality, quality, complexity, reliability, maintainibity, etc. or many more.

Measurement Process:-

There are Five Steps:-

i) **Formulation:-** This perform measurement and develop appropriate metrics for s/w under consideration.

ii) **Collection:-** Collection of Data.

iii) **Analysis:-** To analyse the collected data.

iv) **Interpretation:-** This analyse the matric to attain inside the quality of representation.

v) **Feedback:-** This communicate recommentations derieved from project metrics to the software tree.

5/8/19

Metrics:-

Once measures are collected they are converted into matrics for use.

⇒ **Objectives of Metrics :-**

i) Measuring the size of the S/W Quantitatively

ii) Accessing the level of complexity involved

iii) Accessing the strength of the module by measuring coupling.

iv) Accessing the testing techniques.

v) Determining the date of release of the S/W.

vi) Estimating cost of resources and project schedule.

⇒ **Guidelines for S/W Metrics :-**

i) Simple & Computable.
ii) Consistent.
iii) Programming language Independent
iv) High Quality
v) Easy to Calibrate
vi) Easy to Obtain
vii) Validation
viii) Robust
ix) Valuable

Unit – II

9/8/19

1. Estimation:- It is the process of finding an estimate or approximation, which is a value that can be used for some purpose, even if input data may be incomplete, uncertain or unstable.

→ Estimation is based on ① Past Data / Past Experience

② Available documents / Knowledge
③ Assumptions
④ Identified Risks

ⓐ Software Size Estimation:-

Size may be estimated in terms of KLOC (Kilo lines of Count) or by calculating the number of function points in the software.

→ It depends on user or s/w requirements

ⓑ Effort Estimation:-

The managers estimate efforts in terms of personnel requirement and man-hour required to produce the s/w.

→ For effort estimation, S/W size should be known.

(c) Time Estimation:-

Once size and efforts are estimated, the time required to produce the software can be estimated.

(d) Cost Estimation:-

For estimating project cost, it is required to consider size of S/w, Quality, Hardware, Additional S/w or Tools, License, skilled Personnel, travel involved, Communication, Training & Support.