UNIT-1

Q.1. Explain the stages involved in Project Management life cycle in detail.

Ans: Stages in Project Management Life cycle:

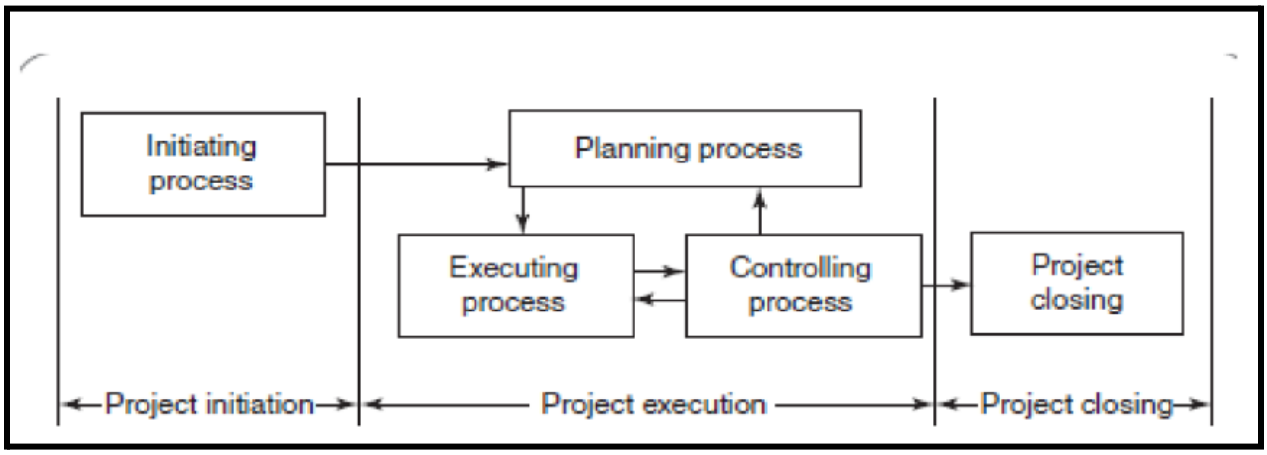


Diagram: Project management life cycle(PMLC)

Project Initiation:

- The main goal is to formally select and start off projects

- Key outputs include:
  - –Assigning the project manager.
  - –Identifying key stakeholders.
  - –Completing the project charter.

Project Planning:
Various plans are made:
– Project plan: Assign project resources and time frames to the tasks.
– Resource plan: List the resources, manpower and equipment that required to execute the project.
– Financial plan: plan for manpower, equipment and other costs.
– Quality plan: Plan of quality targets and control.
– Risk plan: Identification of the potential risks, their prioritization and a plan for the actions that would be taken to contain the different risks.

Project Execution:
• Tasks are executed as per the project plan

• Monitoring and control processes are executed to ensure that the tasks are executed as per plan
• Corrective actions are initiated whenever any deviations from the plan are noticed.

Project Closure:
• Involves completing the release of all the required deliverables to the customer along with the Necessary documentation.
• Subsequently, all the project resources are released and supply agreements with the vendors are terminated and all the pending payments are completed.
• Finally, a post-implementation review is undertaken to analyze the project performance and to list the lessons learnt for use in future projects.

Q.2 . What is Project portfolio management? Explain the key aspects of Project portfolio management.

Ans: Project portfolio management:

Project portfolio management (PPM) refers to a process used by project managers and project management organizations (PMOs) to analyze the potential return on undertaking a project.

The concerns of project portfolio management include:

• Evaluating proposals for projects
• Assessing the risk involved with projects
• Deciding how to share resources between projects
• Taking account of dependencies between projects
• Removing duplication between projects
• Checking for gaps

There are three elements to PPM:

1. Project portfolio definition
– Create a central record of all projects within an organization
– Must decide whether to have ALL projects in the repository or, say, only ICT projects
– Note difference between new product development (NPD) projects and renewal projects e.g. for process improvement

2. Project portfolio management
Actual costing and performance of projects can be recorded and assessed.

3. Project portfolio optimization
Information gathered above can be used achieve better balance of projects e.g. some that are risky but potentially very valuable balanced by less risky but less valuable projects.

Q.3. Define the following terms: i) Net Profit ii) Return on Investment iii) Payback Period iv) Net present value v) Internal rate of return
Ans:

Net profit:

- The net profit of a project is the difference between the total cost and the total income over the life of the project
- Not a very good technique because of no consideration of time

Return on investment (ROI):

$$\textbf{Return on investment (ROI)} = \frac{\textbf{Average annual profit}}{\textbf{Total investment}} \times 100$$

average annual profit = 50,000/5 = 10,000
ROI = 10,000/100,000 X 100 = 10%

Pay back period:

- Payback period is the time taken to break even or payback the initial investments.
- Payback period is a simple calculation of time for the initial investment to return.
- It is the time required to recover the initial cost of an investment.
- The shorter the payback period, the more attractive the investment.

The formula for computing payback period with even cashflows is:

Pay back period = Total outflows / Inflow every year
or
Initial investment / Net annual cash inflows

Net present value:

- Net present value is a tool of Capital budgeting to analyze the profitability of a project or investment.
- It is the sum of the present values of all future amounts.
- Present value is the value which a future amount is worth at present
- It takes into account the profitability of a project and the timing of the cash flows

Formula for NPV :

$$NPV = (\text{Cash flows})/(1+r)^t$$

Cash flows= Cash flows in the time period

r = Discount rate

t = time period

1/(1+r)^t is known as discount factor

Internal rate of return:

- It is the discount rate at which the present value of a project's net cash inflows becomes equal to the present value of its net cash outflows.
  Equation :
  present value of a project's net cash inflows = present value of its net cash outflows.
- In other words, internal rate of return is the discount rate at which a project's net present value becomes equal to zero.

Q.4. What is Project charter in software project management? What are the elements of Project charters?

Ans: Project charter in software project management:

1. A project charter is the statement of scope, objectives and people/team who are participating in a project.
2. A Project Charter is a formal document that authorizes the project team to execute project activities
3. It provides the project manager with the ability to apply organizational resources to project activities.
4. The Project Manager at the project initiation phase writes this document.

While outlining the project objectives, a project charter provides the following major functions:

- A preliminary demarcation of the roles and responsibilities that each individual plays during the proceedings of a project
- Identification of primary stakeholders or project owners and project sponsors
- Outlining of the duties and roles undertaken by a project manager

The project charter contains the following elements:

1. Time: the start date and the deadline for the project.
2. People involved in the project.
3. Outlined objectives and set targets.
4. The return expected from the project.
5. Clearly defined roles and responsibilities of the participants involved.

6. Requirement of resources that will be needed for the objectives to be achieved.
7. Barriers and the risks involved with the project.

Unit -2

Q.1. State Capers Jones Rules of thumb for software estimation.
Ans: Capers Jones Rules of thumb:

Rule 1: SLOC-function point equivalence:
One function point = 125 SLOC for C programs.

Rule 2: Project duration estimation:
Function points raised to the power 0.4 predicts the approximate development time in calendar months.
E.g. 150 FP = 150X125 = 18750 SLOC, Development time = (150)0.4=7.42==8 Months

Rule 3: Rate of requirements creep:
User requirements creep in at an average rate of 2% per month from the design through coding phases.

Rule 4: Defect removal efficiency:
Each software review, inspection, or test step will find and remove 30% of the bugs that are present.

Rule 5: Project manpower estimation:
The size of the software (in function points) divided by 150 predicts the approximate number of personnel required for developing the application.
E.g. 500 FP/150 = 4 personals (approx.)

Rule 6: Software development effort estimation:
The approximate number of staff months of effort required to develop a software is given by the software development time multiplied with the number of personnel required.
Eg. 8 months X (150 FP/ 150) = 8 person-month

Rule 7: Number of personnel for maintenance
Function points divided by 500 predicts the approximate number of personnel required for regular maintenance activities.
E.g. 500/500 = 1 person

Q.2. Explain the five major components of Albrecht function point analysis.IMP

Ans: Components of Albrecht function point analysis:

Albrecht Function point analysis

- This is a top-down method that was devised by Allan Albrecht when he worked for IBM.
- Albrecht was investigating programming productivity and needed some way to quantify the functional size of programs independently of the programming languages in which they had been coded.
- He developed the idea of function points (FPs)
- FP (Function Point) is the most widespread functional type metrics suitable for quantifying a software application.
- It was an attempt to overcome difficulties associated with lines of code as a measure of software size, and to assist in developing a mechanism to estimate effort associated with software development.
- In the world of Function Point Analysis, systems are divided into five large classes and general system characteristics.
- The first three classes or components are External Inputs,External Outputs and External Inquires each of these components transact against files therefore they are called transactions.
- The next two Internal Logical Files and External Interface Files are where data is stored that is combined to form logical information.

Classes or components :

1. Logical interface file (LIF) types – equates roughly to a data store in systems analysis terms. Created and accessed by the target system
2. External interface file types (EIF) – where data is retrieved from a data store which is actually maintained by a different application.
3. External input (EI) types – input transactions which update internal computer files
4. External output (EO) types – transactions which extract and display data from internal computer files. Generally involves creating reports.
5. External inquiry (EQ) types – user initiated transactions which provide information but do not update computer files. Normally the user inputs some data that guides the system to the information the user needs.

External user types	Low complexity	Medium complexity	High complexity
EI	3	4	6
EO	4	5	7
EQ	3	4	6
LIF	7	10	15
EIF	5	7	10

Q.3. Explain the top down approach associated with Parametric Models.
Ans: Top down approach associated with Parametric Models:

1. The top-down approach is normally associated with parametric (or algorithmic) models. These may be explained using the analogy of estimating the cost of rebuilding a house.

2. Analogy/Example: Unless the house-owner is in the building trade he or she is unlikely to be able to calculate the numbers of bricklayer-hours, carpenter-hours, electrician-hours, and so on, required. Insurance companies, however, produce convenient tables where the house-owner can find estimates of rebuilding costs based on such parameters as the number of floors and the floor space of a house. This is a simple parametric model.

3. Project effort relates mainly to variables associated with characteristics of the final system. A parametric model will normally have one or more formulae in the form:

effort = (system size) X (productivity rate)

4. For example, system size might be in the form ‘thousands of lines of code’ (KLOC) and have the specific value of 3 KLOC while the productivity rate was 40 days per KLOC. These values will often be matters of judgment. A model to forecast software development effort therefore has two key components. The first is a method of assessing the amount of the work needed. The second assesses the rate of work at which the task can be done.
productivity = effort/size

5. A more sophisticated way of doing this would be by using the statistical technique least squares regression to derive an equation in the form:
effort = constant, + (size X constant)

6. Some parametric models, such as that implied by function points, are focused on system or task size, while others, such are COCOMO, are more concerned with productivity factors.

7. Top-down estimates steps
• Produce overall estimate using effort driver(s)
• distribute proportions of overall estimate to components

Q.4. Write a note on COCOMO II Model.IMP

Ans: COCOMO II model:

- COCOMO-II is the revised version of the original Cocomo (Constructive Cost Model) and is developed at University of Southern California.
- It is the model that allows one to estimate the cost, effort and schedule when planning a new software development activity.
- COCOMO II incorporates a range of sub-models that produce increasingly detailed software estimates.
- COCOMO II is tuned to modern software life cycles.
- It is the advanced model that estimates the software development effort like Intermediate COCOMO in each stage of the software development life cycle process.
- Produces increasingly accurate estimates.

The 4 sub-models in COCOMO 2 are:

1. **Application composition model.** Used when software is composed from existing parts.

2. **Early design model**. Used when requirements are available but design has not yet started.

3. **Reuse model**. Used to compute the effort of integrating reusable components.

4. **Post-architecture model**. Used once the system architecture has been designed and more information about the system is available.

$$\text{effort} = c \times \text{size}^k$$

• c and k depend on the type of system: organic, semi-detached, embedded
• Size is measured in 'KLOC' ie. Thousands of lines of code

Effort multipliers in COCOMO II Model:

The scale factor effort multipliers are also assessed:

RCPX Product reliability and complexity

RUSE Reuse required
PDIF Platform difficulty
PERS Personnel capability
FCIL Facilities available
SCED Schedule pressure

Q.5. Explain the following with respect to estimation ? Parkinson's Law, Brooks' Law

Or

Over and under-estimating and Software Efforts estimation techniques?

ANS-

Over and under-estimating :

An over-estimate might cause the project to take longer than it would otherwise. This can be explained by the application of two 'laws'.

Parkinson's Law - 'Work expands to fill the time available', which implies that given an easy target staff will work less hard.

Brooks' Law –

- The effort required to implement a project will go up disproportionately with the number of staff assigned to the project.
- As the project team grows in size so will the effort that has to go into management, co-ordination and communication.
- This has given rise, to the notion of Brooks' Law: 'putting more people on a late job makes it later'.
- If there is an over-estimate of the effort required then this might lead to more staff being allocated than are needed and managerial overheads will be increased.
- This is more likely to be of significance with large projects.

Software Efforts estimation techniques

Barry Bohm identified the main ways of deriving estimated of software development efforts as:

1. Algorithmic Models- which use 'effort drivers'to predict effort
2. Expert judgment - where the advice of knowledgeable staff is solicited;

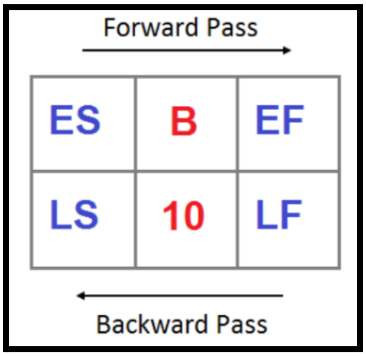
3. Analogy - where a similar, completed, project is identified and its actual effort is used as a basis for the new project;
4. Parkinson - which identifies the staff effort available to do a project and uses that as the 'estimate';
5. Price to win - where the 'estimate' is a figure that appears to be sufficiently low to win a contract;
6. Top-down - where an overall estimate is formulated for the whole project and is then broken down into the effort required for component tasks;
7. Bottom-up - where component tasks are identified and sized and these individual estimates are aggregated.

Unit - 3

Q.1.: With the help of example explain forward pass and backward pass to calculate activity duration in network diagram.(?)

Ans-

1. Forward pass is a technique to move forward through network diagram to determining project duration and finding the critical path or Free Float of the project.
2. Whereas backward pass represents moving backward to the end result to calculate late start or to find if there is any slack in the activity.
3. **Slack (Float)** – The amount of time that a schedule activity may be delayed from its Early Start date without delaying the project finish date



Forward pass :

What is Early Start?

- Early Start (ES) represents the earliest start of an activity considering the dependency preceding task. If an activity is having more than one dependency predecessor, then ES will be the highest Early Finish (EF) of the dependency task.

- Early Start = Maximum (or Highest) EF value from immediate Predecessor(s)

How to apply Forward Pass to calculate Early Finish (EF)?

In order to calculate Early Finish, we use forward pass. Means moving from Early Start towards right to come up with Early Finish of the project.

- Early Finish (EF) = ES + Duration

- If Early Start is 6 days and duration is 10 days, EF = 6 + 10 = 16 Days

Backward Pass

What is Late Finish (LF)?

- Late Finish(LF) is the latest date that the activity can finish without causing a delay to the project completion date.

How to apply Backward Pass to calculate Late Start (LS)?

- In order to calculate Late Start (LS), we apply backward Pass moving from Late Finish and deducting from activity duration.

  LS = LF – Duration

  If Late Finish is 30 days and duration is 10 days, LS = 30 – 10 = 20 Days

Q.2.: Explain Activity planning and activity network in Details?it will cover 2 questions

Ans -

A detailed plan for the project includes a schedules indicating the start and completion time for each activity. This will enable to –

1. Ensure that **appropriate resources** will be available precisely when required.
2. **Avoid** different activities **competing** for the **same resources** at the same time.
3. Produce a **detailed schedule** showing which staff carry out each activity.
4. **Replan** the project during its life to **correct drift** from the target.

Objectives of Activity Planning (FRDM)

1. **Feasibility Assessment** - Is the project possible within required timescale and resource constraint.
2. **Resource Allocation**- The project plan allow us to investigate the relationship between timescale and resource availability.
3. **Detailed costing**- How much will the project cost and when is that expenditure likely to take place.
4. **Motivation**- Providing targets and being seen to monitor achievements against targets is an effective way of motivating staff.

Activity Network :

- Activity network (activity graph) A **graphical method** for showing **dependencies** between tasks (activities) in a project.
- An Activity Network Diagram is a diagram of project activities that shows the **sequential relationships** of activities using **arrows and nodes.**
- An activity network diagram tool is **used extensively** and is necessary for the **identification** of a project's **critical path .**

Activity networks are based on some assumptions:

A project is:

- Composed of a number of activities
- May start when at least one of its activities is ready to start
- Completed when all its activities are completed

An activity :

- Must have clearly defined start and end-points
- Must have resource requirements that can be forecast: these are assumed to be constant throughout the project
- Must have a duration that can be forecast
- May be dependent on other activities being completed first (precedence networks)

Q.3. Explain Bohem's top ten software project risks and the different strategies for reducing it. IMP

Ans: Boehm's top 10 development risks:
Barry Boehm surveyed software engineering project leaders to find out the main risks that they had experienced with their projects. For each risk, some risk reduction techniques has been suggested.

Risk	*Risk reduction techniques*
Personnel shortfalls	Staffing with top talent; job matching; teambuilding; training and career development; early scheduling of key personnel
Unrealistic time and cost estimates	Multiple estimation techniques; design to cost; incremental development; recording and analysis of past projects; standardization of methods
Developing the wrong software functions	Improved software evaluation; formal specification methods; user surveys; prototyping; early user manuals
Developing the wrong user interface	Prototyping; task analysis; user involvement
Gold plating	Requirements scrubbing, prototyping, design to cost
Late changes to requirements	Change control, incremental development
Shortfalls in externally supplied components	Benchmarking, inspections, formal specifications, contractual agreements, quality controls
Shortfalls in externally performed tasks	Quality assurance procedures, competitive design etc
Real time performance problems	Simulation, prototyping, tuning

Q.4.. Define the following terms: i) critical path ii) float ii) free float iv) interfering float vi)Feeding Buffer vii)Project Buffer

Ans:

Critical path:

- It is the path through network with zero floats.
- Critical path is a common type of technique used by project managers when it comes to resource leveling.
- Any delay in an activity on this path will delay whole project
- This method calculates the **minimum completion time** for a project along with the possible start and finish times for the project activities.
- The duration of the critical path is the sum of the activities' durations along the path.
- Thus, the critical path can be defined as the longest possible path through the "network" of project activities,

Float: Float = Latest finish - Earliest start - Duration
Float can also be calculated as the difference between the earliest and latest start dates for an activity or the difference between the earliest and latest finish dates.

Free float:

- Free Float = Free float is measured by subtracting the early finish (EF) of the activity from the early start (ES) of the successor activity.
- Free Float represents the amount of time by which an activity may be delayed without affecting any subsequent activity.

Interfering float:

Interfering float = total float – free float

Project Buffer -

1. Put a project buffer at the end of the critical chain with duration 50% of sum of comfort zones of the activities on the critical chain.
2. This buffer is placed between the last task and the project completion date as a non-activity buffer, and it acts as a contingency for the critical chain activities. Any delay on the critical chain will eat this buffer and the project completion date will not change.
3. If any activity finishes early, the gain will be added to this buffer.

Feeding Buffer -

1. Where subsidiary chains of activities feed into critical chain, add feeding buffer
2. Duration of feeding buffer 50% of sum of comfort zones of activities in the feeding chain
3. Where there are parallel chains, take the longest and sum those activities

Q.5. Explain Critical chain Method in detail?

ANS -

Critical Chain Approach

- The critical chain is “the longest path in the network diagram considering activity interdependence and resource constraints.”
- The general steps in the Critical Chain approach are explained in the following sections:

  A) Deriving ‘most likely’ activity durations.

  B) Using latest start dates for activities.

  C) Inserting project and feeder buffers.

  D) Project execution.

- It’s used to prepare the project schedule when limited or restricted resources are available.
- The goal of CCM is eliminating project schedule delays due to uncertainties, overestimation of task duration and wasted internal buffers.

Critical Chain Method (CCM) :

1. The Critical Chain Method is an updated form of the Critical Path Method.
2. Here, we consider limited resource availability while developing the project schedule.
3. In the Critical Chain Method, you use a buffer instead of float.
4. Critical path is a particular case of the critical chain when the project has unlimited resources.

Critical Chain Method (CCM) - Buffers :

Critical chain management has two buffers.

Project Buffer -

1. Put a project buffer at the end of the critical chain with duration 50% of sum of comfort zones of the activities on the critical chain.

2. This buffer is placed between the last task and the project completion date as a non-activity buffer, and it acts as a contingency for the critical chain activities. Any delay on the critical chain will eat this buffer and the project completion date will not change.

3. If any activity finishes early, the gain will be added to this buffer.

Feeding Buffer -

1. Where subsidiary chains of activities feed into critical chain, add feeding buffer
2. Duration of feeding buffer 50% of sum of comfort zones of activities in the feeding chain
3. Where there are parallel chains, take the longest and sum those activities

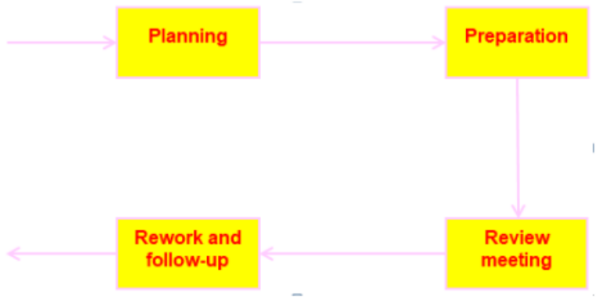
Unit 4

Q.1 . Explain review process model with the help of diagram.
Ans:
Review Process Model:
• Review of work products is an important mechanism for monitoring the progress of a project and ensuring the quality of the work products.
• Testing is an effective defect removal mechanism.
– However, testing is applicable to only executable code.
– Review is applicable to all work products.

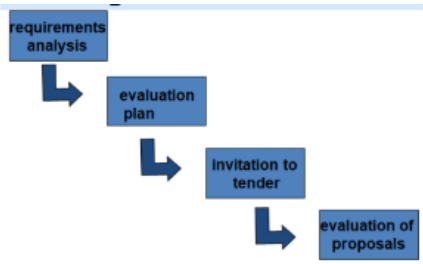
Utility or benefit of Review:
• A cost-effective defect removal mechanism.
• Review usually helps to identify any deviation from standards.
• Reviewers suggest ways to improve the work product
• A review meeting often provides learning opportunities to not only the author of a work product, but also the other participants of the review meeting.
• The review participants gain a good understanding of the work product under review, making it easier for them to interface or use the work product in their work.

Review Roles:
• Moderator: Schedules and convenes meetings, distributes review materials, leads and moderates review sessions.
• Recorder: Records the defects found and the time and effort data.
• Reviewers: The persons who performs the review.

Review Process:



Q.2. : Explain the stages in Contract Placement
Ans: Stages in Contract Placement:



Step 1 - Requirements Analysis:
Requirements document: sections
• introduction
• description of existing system and current environment
• future strategy or plans
• system requirements -
– mandatory/desirable features
• deadlines
• additional information required from bidders
• Requirements should include
– functions in software, with necessary inputs and outputs
– other applications with which software is to be compatible
– quality requirements e.g. response times

Step 2 - Evaluation plan:
• Need to assess value for money (VFM) for each desirable feature
• VFM approach an improvement on previous emphasis on accepting lowest bid
• Example:
– feeder file saves data input

Step 3 - Invitation to tender (ITT):
• Note that bidder is making an offer in response to ITT
• acceptance of offer creates a contract
• Customer may need further information
• Problem of different technical solutions to the same problem

Step 4 – Evaluation of Proposals:
The purpose of evaluation includes:
• Scrutiny of the proposal document
• Interviewing supplier's representatives
• Demonstration
• Site visit
• Practical test

Q3. Write a note on ethical and professional concerns as a member of any organization.

Ans:

Q.4.: Discuss the factors of job satisfaction given by oldham-Hackman . also state the method of improving motivation .

ANS -

Oldham and Hackman suggest that the satisfaction that a job gives is based on five factors,

The first three factors make the job'meaníngful'to the person who is doing it:

1. **skill variety:** the number of different skills that the job holder has the opportunity to exercise;
2. **task identity:** the degree to which your work and its results are identifiable as belonging to you;
3. **task significance:** the degree to which your job has an influence on others.

The other two factors are:

4. **autonomy:** the discretion you have about the way that you do the job;
5. **feedback:** the information you get back about the results of your work.

Oldham and Hackman also noted that both the job holders' personal growth needs and their working environment influenced their perception of the job. Some writers have pointed out that if people are happy with their work for other reasons, they are likely to rate it higher on the Oldham-Hackman dimensions anyway.

Method of improving motivation :

1. **Set specific goals** These goals need to be demanding and yet acceptable to staff. involving staff in the setting of goals helps to gain acceptance for them.

2. **Provide feedback** Not only do goals have to be set but staff need regular feedback about how they are progressing.

3. **Consider job design** Jobs can be altered to make them more interesting and give staff more feeling of responsibility.

Two measures are often used to enhance job design - job enlargement and job enrichment :

1. **Job enlargement-**the person doing the job carries out a wider variety of activities. It is the opposite of increasing specialization.
2. **Job enrichment-** The job holder carries out tasks that are normally done at a managerial or supervisory level.

Q.5. Define the any two terms and explain with example: i) Scheduling Variance ii) Cost Variance iii) Earned Value iv) Time Variance

Ans-

Schedule Variance: The schedule variance is measured in cost as EV-PV and indicates the degree to which the value completed work differs from that planned. Say, for example, that work with a PV of 40000 should have been completed by now. In fact, some of that work has not been done so that EV is only 35000. The SV would therefore is 35000 – 40000 = 5000. A negative SV means the project is behind schedule.

Time variance (TV) – difference between time when specified EV should have been reached and time it actually did reach.

For example, say an EV of £19000 was supposed to have been reached on 1st April and it was actually reached on 1st July then TV = - 3 months

Earned value (EV) or Budgeted cost of work performed (BCWP) – total of PVs for the work completed at this time
Eg. :

Cost variance (CV): This is calculated as EV — AC and indicates the difference between the earned value or budgeted cost and the actual cost of completed work. Say that when the SV above was calculated as –

EV = $160,000 AC = $145,000
What does this mean? What is the CV?

CV = EV – AC = + $15,000 we are under budget

Q.6. What is fixed price contract? List the advantages and disadvantages of fixed price contract.

Ans: Fixed price contract:

• A price is fixed when the contract is signed
• If no change in contract ,this is the price they pay on completion
• Once the development is under way the customer cannot change their requirements without renegotiating the price of the contract.

Advantages:

• known expenditure
• supplier motivated to be cost-effective.

Disadvantages:

• supplier will increase price to meet contingencies
• difficult to modify requirements
• cost of changes likely to be higher
• threat to system quality

When competing for work, there will be pressure on the suppliers to reduce prices. Once a contract has been won and signed, the contractor is in a stronger negotiating position when it comes to negotiating the price of additional work as the customer is now locked in.

UNIT-5

Q.1.What are the different types of Team Structure?

Ans -

Team Structure:

• We consider only three team structures:
- **– Democratic,**
- **– Chief programmer,**
- **– Mixed team**

- **Chief programmer**

1. In this team organization, a senior engineer provides the technical leadership and is designated as the chief programmer.
2. The chief programmer partitions the task into small activities and assigns them to the team members.
3. since team-members work under the constant supervision of the chief programmer. This also inhibits their original thinking.
4. The chief programmer team is subject to single point failure since too much responsibility and authority is assigned to the chief programmer.

- **Democratic**

1. Does not enforce any formal team hierarchy.
2. Decisions are taken based on discussions,
   – any member is free to discuss with any other member
3. Since a lot of debate and discussions among the team members takes place,
   – for large team sizes significant overhead is incurred

- **Mixed team**

1. This team organization incorporates both hierarchical reporting and democratic set up.
2. The mixed control team organization is suitable for large team sizes.

3. The democratic arrangement at the senior engineers level is used to decompose the problem into small parts.
4. Each democratic setup at the programmer level attempts solution to a single part.
5. Thus, this team organization is mostly suited to handle large and complex programs

Q.2 What is CMM (Capability Maturity Model)? What are the various levels of CMM?

Ans -

- CMM has been developed at software engineering institute
- Software organizations are placed at one of the five levels of process maturity to indicate the quality of their software production practices.

  Level 1: Initial
  Level 2: Repeatable
  Level 3: Defined
  Level 4: Managed
  Level 5: Optimized

Level 1: Initial

- The procedures followed tend to be haphazard. Any organization is at this level by default.
- Software development processes are not defined
- Organization operates Without any formalized process or project plans

Level 2: Repeatable

- Basic project management practices are followed
  – Size and cost estimation techniques: Function point analysis, COCOMO, etc.

Level 3: Defined

- Defined and documented.
- The organization has defined the way in which each task in the software development life cycle is to be done.

Level 4: Managed

- The products and processes involved in software development are subjected to measurement and control.
- Results of measurement used to evaluate project performance

Level 5: Optimizing

- Improvement in procedures are designed and implemented using the data gathered from the measurement process.