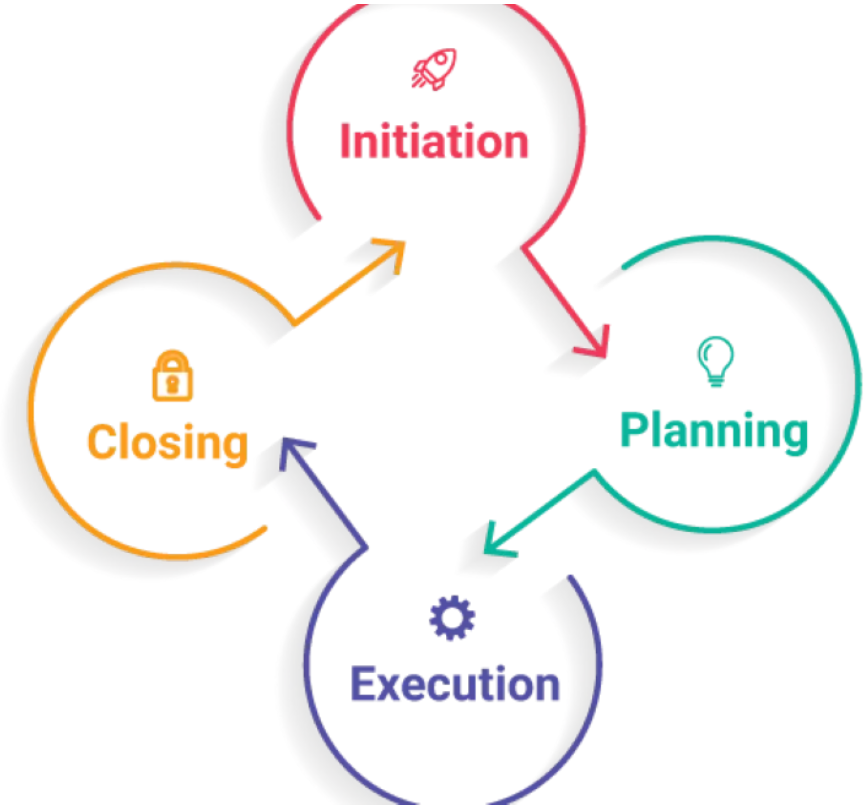
Q1)Define Project And Project LifeCycle

=>



A project is a combination of interrelated activities to achieve a specific objective within a schedule, budget, and quality. It involves the coordination of group activity, wherein the manager plans, organizes, staffs directs, and controls to achieve an objective, with constraints on time, cost, and performance of the end product. Project management is the combination of project and management.Planning is the strong keys to make the project more effective and well utilization of resources to achieve the goal.

The Project Lifecycle is the sequence of phases through which a project progresses. It includes initiation, planning, execution, and closure. Learn more.

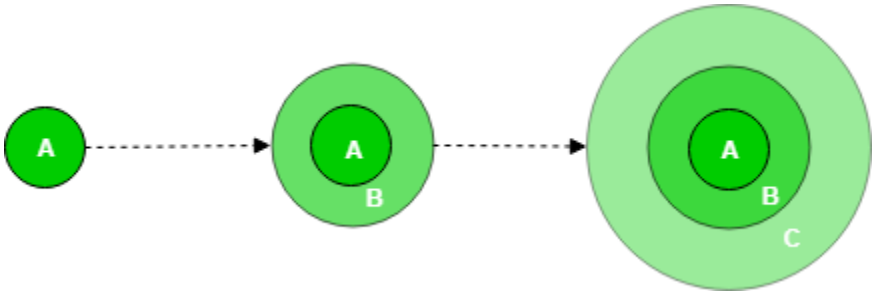
The Project Lifecycle is the sequence of phases through which a project progresses. The number of phases and sequence of the cycle may vary based on the company and the type of project undergone. As part of a project, however, they should have a definite start and end, and they are constrained by time. The lifecycle provides the basic foundation of the actions that has to be performed in the project, irrespective of the specific work involved.

The Project Lifecycle typically involves these four stages:

1. **Project Initiation** – this is the start of the project. It may involve many sub-activities including: a feasibility study, identifying the scope, identifying deliverables, identifying project stakeholders, developing a business case, creating a statement of work, and possibly initial costs, price, and timeline for work to be done.
2. **Project Planning** – Once the project is approved from the initiation phase, it moves into planning. This phase involves creating a project plan, including the tasks, schedule, resources, and constraints on the project. The budget for the project is also created in this phase. In addition, risk should be anticipated and identified at this stage, as well as mitigation plans.
3. **Project Execution** – This phase is where the work gets done. Task owners begin work and the project manager oversees that tasks are done in a timely manner and workflow continues smoothly. Monitoring and Controlling (managing the work and financials) are a big part of this phase, as issues will always arise and require quick adjustments as the project progresses.
4. **Project Closure** – Once the team has completed all the tasks, and the project owner signs off that all deliverables are complete, the project is closed. Any documentation is handed over to the project owner and if required to an ongoing maintenance organization. The project is then analyzed for performance to determine whether the project's goals were met (tasks completed, on time and on budget).

Q2)Explain Incremental Model In Detail..
=>
The incremental process model is also known as the Successive version model.

First, a simple working system implementing only a few basic features is built and then that is delivered to the customer. Then thereafter many successive iterations/ versions are implemented and delivered to the customer until the desired system is released.



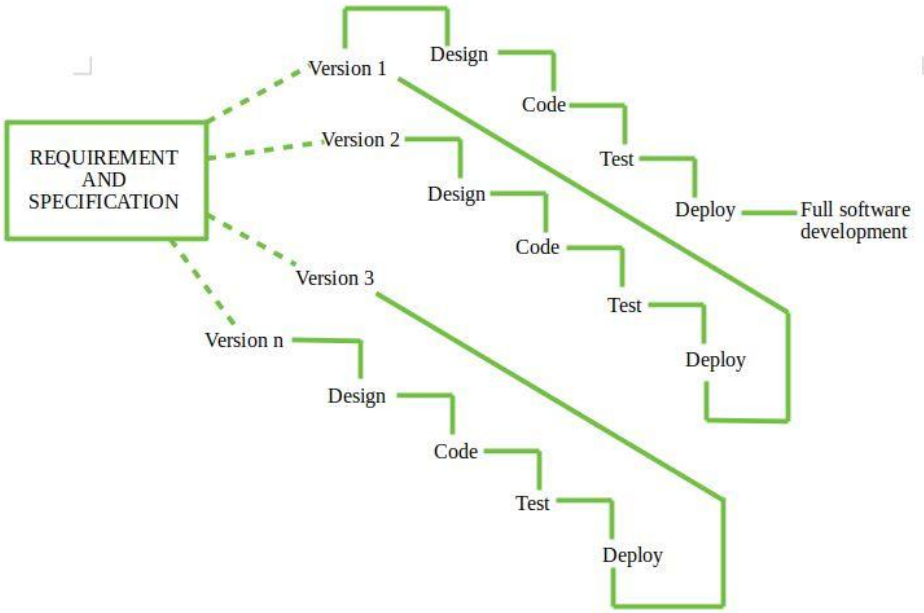
A, B, and C are modules of Software Products that are incrementally developed and delivered.

Life cycle activities:

Requirements of Software are first broken down into several modules that can be incrementally constructed and delivered. At any time, the plan is made just for the next increment and not for any kind of long-term plan. Therefore, it is easier to modify the version as per the need of the customer. The Development Team first undertakes to develop core features (these do not need services from other features) of the system.

Once the core features are fully developed, then these are refined to increase levels of capabilities by adding new functions in Successive versions. Each incremental version is usually developed using an iterative waterfall model of development.

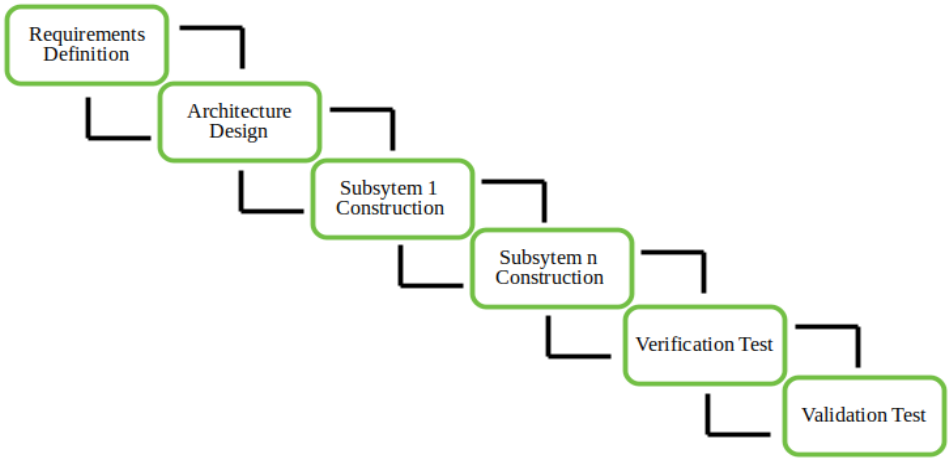
As each successive version of the software is constructed and delivered, now the feedback of the Customer is to be taken and these were then incorporated into the next version. Each version of the software has more additional features than the previous ones.



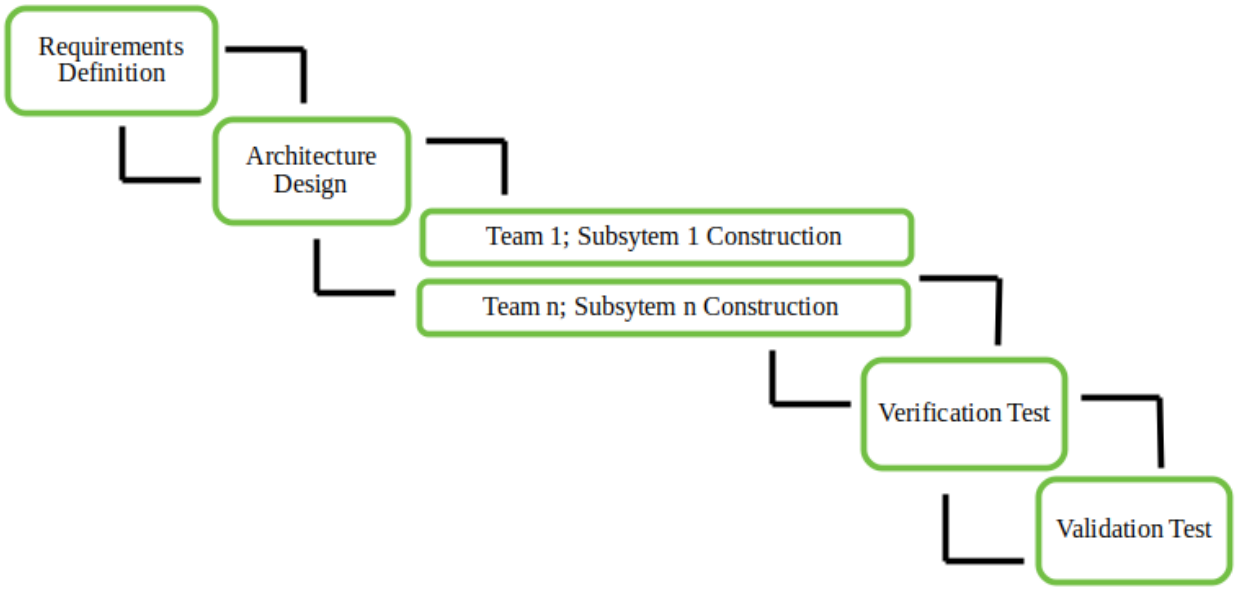
After Requirements gathering and specification, requirements are then split into several different versions starting with version 1, in each successive increment, the next version is constructed and then deployed at the customer site. After the last version (version n), it is now deployed at the client site.

Types of Incremental model:

1. Staged Delivery Model: Construction of only one part of the project at a time.



2. Parallel Development Model – Different subsystems are developed at the same time. It can decrease the calendar time needed for the development, i.e. TTM (Time to Market) if enough resources are available.



When to use this:

1. Funding Schedule, Risk, Program Complexity, or need for early realization of benefits.

2. When Requirements are known up-front.

3. When Projects have lengthy development schedules.

4. Projects with new Technology.

- Error Reduction (core modules are used by the customer from the beginning of the phase and then these are tested thoroughly)
- Uses divide and conquer for a breakdown of tasks.
- Lowers initial delivery cost.
- Incremental Resource Deployment.

5. Requires good planning and design.

6. The total cost is not lower.

7. Well-defined module interfaces are required.

Characteristics of an Incremental model –

- System development is divided into several smaller projects.
- To create a final complete system, partial systems are constructed one after the other.
- Priority requirements are addressed first.
- The requirements for that increment are frozen once they are created.

Advantages-

1. Prepares the software fast.
2. Clients have a clear idea of the project.
3. Changes are easy to implement.
4. Provides risk handling support, because of its iterations.
5. Adjusting the criteria and scope is flexible and less costly.
6. Comparing this model to others, it is less expensive.
7. The identification of errors is simple.

Disadvantages-

1. A good team and proper planned execution are required.
2. Because of its continuous iterations the cost increases.
3. Issues may arise from the system design if all needs are not gathered upfront throughout the duration of the program lifecycle.
4. Every iteration step is distinct and does not flow into the next.
5. It takes a lot of time and effort to fix an issue in one unit if it needs to be corrected in all the units.

Q3 -A) Define Requirement Enginearing . Explain any two requirement elicitation techniques.

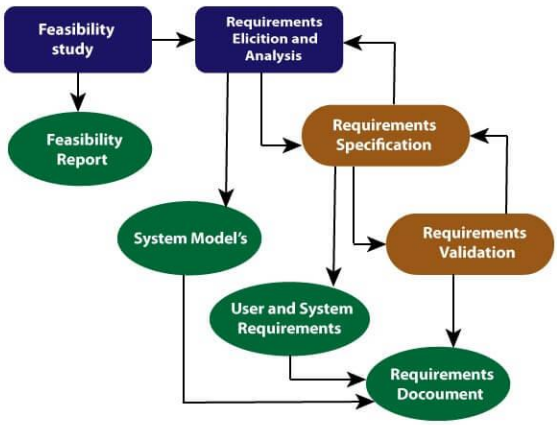
Requirement Engineering

Requirements engineering (RE) refers to the process of defining, documenting, and maintaining requirements in the engineering design process. Requirement engineering provides the appropriate mechanism to understand what the customer desires, analyzing the need, and assessing feasibility, negotiating a reasonable solution, specifying the solution clearly, validating the specifications and managing the requirements as they are transformed into a working system. Thus, requirement engineering is the disciplined application of proven principles, methods, tools, and notation to describe a proposed system's intended behavior and its associated constraints.

Requirement Engineering Process

It is a four-step process, which includes -

1. Feasibility Study
2. Requirement Elicitation and Analysis
3. Software Requirement Specification
4. Software Requirement Validation
5. Software Requirement Management



Requirement Engineering Process

There are a number of requirements elicitation methods. Few of them are listed below –

1. Interviews
2. Brainstorming Sessions
3. Facilitated Application Specification Technique (FAST)
4. Quality Function Deployment (QFD)
5. Use Case Approach

The success of an elicitation technique used depends on the maturity of the analyst, developers, users, and the customer involved.

1. Interviews:

Objective of conducting an interview is to understand the customer's expectations from the software.
It is impossible to interview every stakeholder hence representatives from groups are selected based on their expertise and credibility.

Interviews maybe be open-ended or structured.

1. In open-ended interviews there is no pre-set agenda. Context free questions may be asked to understand the problem.
2. In structured interview, agenda of fairly open questions is prepared. Sometimes a proper questionnaire is designed for the interview.

2. Brainstorming Sessions:

- It is a group technique
- It is intended to generate lots of new ideas hence providing a platform to share views
- A highly trained facilitator is required to handle group bias and group conflicts.
- Every idea is documented so that everyone can see it.
- Finally, a document is prepared which consists of the list of requirements and their priority if possible.

Q3 -B) Explain any two techniques used for software quality control with suitable example ?

Software quality control involves various techniques and processes to ensure that software meets specified requirements and is free from defects. Two commonly used techniques for software quality control are:

1. Static Testing:

- Description: Static testing is a technique that involves reviewing and evaluating the software documentation and source code without actually executing the program. The goal is to identify defects early in the development process, before the code is executed.

- Example: Code reviews and inspections are common static testing techniques. In a code review, developers or a team of developers examine the source code to identify issues such as coding standards violations, logical errors, and potential security vulnerabilities. This process helps catch defects early in the development cycle, reducing the cost of fixing issues later on. For example, a development team may conduct a peer review of a module's source code before integrating it into the larger software system.

2. Dynamic Testing:

- Description: Dynamic testing involves executing the software and observing its behavior to identify defects. This technique is used to validate that the software functions correctly and meets the specified requirements during runtime.

- Example: Unit testing is a common dynamic testing technique. In unit testing, individual components or units of the software are tested in isolation to ensure that each unit performs as intended. For example, a developer might write and execute unit tests for a specific function within a software module. If the function is supposed to calculate the sum of two numbers, the unit test would provide known input values and verify that the output matches the expected result. Automated testing tools are often used for dynamic testing to streamline the process and ensure thorough test coverage.

By combining both static and dynamic testing techniques, software development teams can enhance their ability to deliver high-quality software that meets user expectations and minimizes the risk of defects in production.

Q5-A) explain Procurement Management

Project procurement management is the creation and maintenance of relationships with external resources needed to complete a project. A project procurement manager communicates with vendors to buy, rent or contract products and services needed to achieve project objectives. Most often, the selection of vendors occurs after they have placed bids to partner with businesses seeking their products or services. A project procurement manager then determines which bid and partnerships are most beneficial to their objectives. Further negotiation may take place to ensure fair representation of both party's interests.

Who uses project procurement management?

Project procurement management may be necessary for a variety of industries where projects requiring outsourced materials or services occur. The following industries commonly use project procurement management to meet their project objectives:

- Construction
- Manufacturing
- Engineering
- Technology
- Finance
- Healthcare

Benefits of Project Procurement Management

There are several benefits of adding procurement management to projects of varying size and complexity, including:

Increased quality

Project procurement management allows organizations to negotiate detailed service contracts to best suit their needs and could increase the certainty of high-quality goods and services. An increase in certainty of quality could help stakeholders feel more confident in a project's success and may result in future investments and funding.

Decreased risk

The process of drafting and negotiating service contracts can help organizations more clearly understand their vendor options by outlining costs, processes and service quality. This can help to decrease the risk of violation of contract terms which could delay or negatively impact a project's success.

Controlled cost

Carefully negotiated procurement terms are a great way to reduce costs and control spending. Procurement management may help organizations better understand the costs necessary to complete a project and may help them select products and services from vendors that best fit their needs without exceeding their budget.

Processes in project procurement management

There are four key processes involved in product procurement management:

1. Planning procurement

Planning procurement involves a series of steps that help determine which resources an organization needs for project completion and the extent of its budget. Project procurement managers often consider the following aspects when planning for procurement:

- The materials and resources required to complete the project
- The materials and resources they already have and which need to be outsourced
- Contract requirements for outside purchases
- Delivery date requirements
- Key project milestones and their deadlines
- Legal terms and conditions
- Industry safety standards of materials and resources
- Researching providers and vendors
- Criteria for partnerships

Most often, managers use a written document as a project procurement plan that addresses the above considerations and details any other important information, such as how to handle changes in delivery dates or contract terms.

2. Conducting procurement

After planning for procurement, project procurement managers assess bids from vendors and select partnerships based on their project needs. Any vendor negotiations often take place during this phase of procurement and all involved parties sign the agreed-upon contracts. Project procurement managers may also make payments for products and services at this time.

3. Controlling procurement

Once contracts become active, procurement control and management are important parts of maintaining partnerships with vendors and ensuring the services and products function as they're intended throughout the course of the project. Controlling procurement often includes:

- Evaluating regular internal status updates
- Reviewing contractor agreements
- Reviewing progress and performance updates from vendors
- Conducting inspections and audits
- Assessing work orders
- Issuing additional payment as necessary

4. Closing procurement

Closing procurement involves all necessary steps in ending a partnership or contract. This often involves a review of the work or services completed, renegotiation of any changes to original contract terms and confirmation of payments issued and received. Organizations may also file a formal release of liability upon procurement closing. This contract confirms that the vendor has fulfilled the terms of the original contract and is no longer responsible for any additional involvement in the project.

Project manager responsibilities during procurement

Throughout project procurement, there are several responsibilities of project managers. These may vary depending on the project and its objectives, but there are five common duties of project procurement managers:

1. Project initiation

A project procurement manager may begin the procurement process during project initiation. They often communicate with management, executives and members of the team assigned to the project to discuss objectives and timelines. This may also include discussing the budget, internal capabilities and high-level expectations.

2. Procurement planning

A procurement plan is an important part of aligning expectations and organizing processes. During procurement planning, project managers often create lists and documents that outline necessary resources and materials, timelines and milestones and potential means of contacting vendors. They may also advertise their need for contractors through classified ads, online networking sites and through their professional networks.

Project procurement managers may also work with management and executives to discuss vendor requirements and post-procurement activities.

3. Stakeholder coordination

Projects often involve several stakeholders, including business owners, management, executives, project teams, liability professionals and consultants. The project procurement manager is responsible for coordinating these teams and individuals to ensure mutual understanding of project goals and objectives. To do this, may they identify all important stakeholders and engage them at appropriate times throughout the project.

4. Vendor coordination

Often, multiple vendors may engage in a single project. It's important that project procurement managers effectively coordinate the efforts of multiple contractors. During this process, a project manager may create vendor schedules of collective meetings or demonstrations and facilitate communication between contractors when necessary.

5. Communication of progress

As a project evolves and progresses, a project procurement manager communicates these changes and advancements to all stakeholders involved. This can help to ensure that all teams and individuals working on the project remain informed on deadlines and schedule changes.

Project procurement managers often determine which information is necessary to communicate, collect that information from stakeholders and vendors and circulate it to the appropriate parties. They may also send weekly email updates to inform all team members of current project statuses.

Q6-A) Explain Formal Technical Review FTR. State how it helps it improve the quality of software development

=>
A Formal Technical Review (FTR) is a systematic examination of a software product conducted by a team of individuals with the purpose of identifying and fixing defects early in the development process. FTR is a type of static testing, meaning it is performed without executing the code. The review process involves a structured discussion and analysis of the software artifacts, such as requirements, design documents, and source code.

Key components of a Formal Technical Review include:

1. Preparation:
   - Selection of Artifacts: The team selects the software artifacts to be reviewed. These could include requirements specifications, design documents, source code, or test plans.
   - Distribution of Materials: The selected artifacts are distributed to the review team members well in advance of the scheduled review meeting.

2. Individual Review:
   - Individual Preparation: Review team members individually examine the assigned materials, looking for defects, inconsistencies, or areas of improvement.
   - Documentation of Findings: Each reviewer documents their findings, such as identified defects or suggestions for improvements.

3. Review Meeting:
   - Discussion: The review team meets to discuss the findings. The author of the document being reviewed may clarify points, and the team collectively addresses identified issues.
   - Decision-Making: Decisions are made regarding the acceptance of the document, and action items are assigned to address identified issues.

4. Rework:
   - Correction of Defects: The author of the document incorporates the feedback from the review, addressing identified defects and making necessary improvements.
   - Re-submission: The revised document is resubmitted for further review if required.

5. Follow-Up:
   - Verification of Changes: The review team may conduct a follow-up review to verify that the identified issues have been addressed appropriately.

How Formal Technical Review Improves Software Quality:

1. Early Defect Detection:
   - FTR allows for the early identification of defects in various software artifacts, preventing the propagation of errors to later stages of the development process.

2. Knowledge Sharing:
   - FTR facilitates knowledge sharing among team members. It ensures that the entire team has a common understanding of the software artifacts, requirements, and design decisions.

3. Consistency and Standards Compliance:
   - By systematically reviewing documentation and code against established standards and guidelines, FTR helps ensure that the software follows a consistent design and adheres to coding standards.

4. Continuous Improvement:
   - FTR fosters a culture of continuous improvement. Through the review process, teams learn from their mistakes, identify best practices, and refine their development processes.

5. Reduced Rework Costs:
   - Early detection and correction of defects in the review phase reduce the cost and effort associated with fixing issues in later stages of development or after the software has been deployed.

6. Enhanced Communication:
   - FTR promotes effective communication among team members. The review meetings provide a forum for discussing design decisions, clarifying requirements, and addressing concerns.

In summary, Formal Technical Review is a structured and systematic approach to improving software quality by detecting and addressing defects early in the development process, promoting collaboration among team members, and ensuring adherence to standards and best practices.

Q 6 - B) Explain Feasibility Analysis Thechniques

=> **Feasibility Study** in Software Engineering is a study to evaluate feasibility of proposed project or system. Feasibility study is one of stage among important four stages of Software Project Management Process. As name suggests feasibility study is the feasibility analysis or it is a measure of the software product in terms of how much beneficial product development will be for the organization in a practical point of view. Feasibility study is carried out based on many purposes to analyze whether software product will be right in terms of development, implantation, contribution of project to the organization etc.

Types of Feasibility Study :

The feasibility study mainly concentrates on below five mentioned areas. Among these Economic Feasibility Study is most important part of the feasibility analysis and the Legal Feasibility Study is less considered feasibility analysis.

1. **Technical Feasibility** –
   In Technical Feasibility current resources both hardware software along with required technology are analyzed/assessed to develop project. This technical feasibility study gives report whether there exists correct required resources and technologies which will be used for project development. Along with this, feasibility study also analyzes technical skills and capabilities of technical team, existing technology can be used or not, maintenance and up-gradation is easy or not for chosen technology etc.
2. **Operational Feasibility** –
   In Operational Feasibility degree of providing service to requirements is analyzed along with how much easy product will be to operate and maintenance after deployment. Along with this other operational scopes are determining usability of product, Determining suggested solution by software development team is acceptable or not etc.
3. **Economic Feasibility** –
   In Economic Feasibility study cost and benefit of the project is analyzed. Means under this feasibility study a detail analysis is carried out what will be cost of the project for development which includes all required cost for final development like hardware and software resource required, design and development cost and operational cost and so on. After that it is analyzed whether project will be beneficial in terms of finance for organization or not.
4. **Legal Feasibility** –
   In Legal Feasibility study project is analyzed in legality point of view. This includes analyzing barriers of legal implementation of project, data protection acts or social media laws, project certificate, license, copyright etc.

Overall it can be said that Legal Feasibility Study is study to know if proposed project conform legal and ethical requirements.

5. **Schedule Feasibility** –
   In Schedule Feasibility Study mainly timelines/deadlines is analyzed for proposed project which includes how many times teams will take to complete final project which has a great impact on the organization as purpose of project may fail if it can't be completed on time.

Aim of feasibility study :

- the overall objective of the organization is covered and contributed by the system or not.
- the implementation of the system be done using current technology or not.
- can the system be integrated with the other systems that already exist

Feasibility Study Process :
The below steps are carried out during the entire feasibility analysis.

1. Information assessment
2. Information collection
3. Report writing
4. General information

Need of Feasibility Study :
A feasibility study is so important to stage of the Software Project Management Process as after completion of feasibility study it gives a conclusion of whether to go ahead with proposed project as it is practically feasible or to stop proposed project here as it is not right/feasible to develop or to think/analyze about proposed project again.

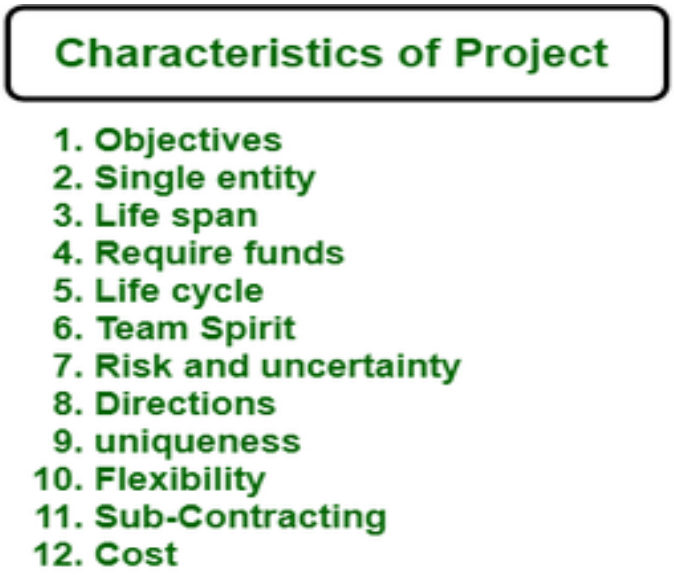
Along with this Feasibility study helps in identifying risk factors involved in developing and deploying system and planning for risk analysis also narrows the business alternatives and enhance success rate analyzing different parameters associated with proposed project development.

1. What is Project. Explain different characteristics of Project.

A project is a combination of interrelated activities to achieve a specific objective within a schedule, budget, and quality. It involves the coordination of group activity, wherein the manager plans, organizes, staffs directs, and controls to achieve an objective, with constraints on time, cost, and performance of the end product. Project management is the combination of project and management.

Planning is the strong keys to make the project more effective and well utilization of resources to achieve the goal. In this, we will focus on characteristics of the project like how objectives are important for achieving the goal, the total time duration of the project, calculated risk, and uncertainty of the project, the total estimated cost of the project, etc. are essential characteristics of the project and will discuss some other characteristics of the project like team spirit, require funds, directions, uniqueness, flexibility, and sub-contracting, etc. Let's discuss it one by one.

Characteristics of a Project :
Projects are **not homogeneous**. Each project is different in itself. The distinctive characteristics of a project are as follows.



1. **Objectives –**
   Every project is started with some objective or goal viz. time, budget, quality, and quantity, when objectives are fulfilled project cause existing. You can initially define the objectives of the project what actually need to achieve. Objectives are the key characteristics of the project where you will see the progress of the project and time to time analysis will show you the result of

how much you have achieved.

2. **Single entity** –
   A project is one whole thing. This means that in a project although different people contribute still is recognized as a single entity. The teams are often specifically assembled for a single project.

3. **Life Span** –
   No project can be ceaseless and indefinite. It must have one and beyond which it cannot proceed. Every project is invariably time-bound. At the time of planning, you will see the time phase of the project where the team can work independently on the project modules. Let's consider an example project that is divided into three modules let's say A, B, and C. If the total time span of a project is 5 months then you can set the time span for modules independently like A can complete in 2 months and also B can complete in 2 months and C can complete in 1 month as per requirement.

4. **Require funds** –
   Every project needs funds to reach the endpoint. Without adequate funds, no project can be successfully implemented. Cost estimation is one of the essential factors for any organization. So, calculating in advance the required funds for the project will be very impactful.

5. **Life Cycle** –
   Each project has a life cycle with different stages like start, growth, maturity, and decay. A project has to pass through different stages to get itself completed. Let's consider an example where the project is related to software development then you can say SDLC (Software Development lifecycle) will be the life cycle of the project where you will see many stages like planning, defining, designing, building, testing, and deployment, etc.

6. **Team Spirit** –
   Team spirit is required to get the project completed because the project constitutes different members having different characteristics and from various disciplines. But to achieve common goal harmony, missionary zeal, team spirit is necessary.

7. **Risk and Uncertainty** –
   The project is generally based on forecasting. So risk and uncertainty are

always associated with projects. There will be a high degree of risk in those project which are not properly defined. Only the degree of control over risk and uncertainty varies with the project being conceived based on information available.

8. **Directions –**
   Project is always performed according to the directions given by the customers with regard to time, quality and quantity, etc. The convenience of the supply sides of economics such as labor availability ore resources and managerial talent etc. are all secondary concerns, primary being the customer requirement.

9. **Uniqueness –**
   Each project is unique in itself, and it's having own features. No two projects are similar even if the type of organization is the same. The uniqueness of the project can measure by considering the many factors like objectives, features of the project, application of the project, etc.

10. **Flexibility –**
    Change and project are synonymous. A project sees many changes throughout its life span. These changes can make projects more dynamic and flexible.

11. **Sub-Contracting –**
    Sub-contracting is a subset of every project and without which no project can be completed unless it is a proprietary firm or tiny in nature. The more complexity of a project the more will be the extent of contracting. Every project needs the help of an outsider consultant, engineer, or expert in that field.

12. **Cost –**
    If the quality of the project is to be changed there could be an impact on the cost of the project. The cost could increase if more resources are required to complete the project quicker.

Q4)Explain Modern Approaches to Leadership & Leadership Styles

The process of influencing the behaviour of people towards the achievement of organisational goals is known as Leadership. It indicates the ability of an individual to maintain good interpersonal relations with followers and motivate them to contribute to achieving organisational objectives. An individual who has the attributes of leadership is known as a leader.

The behaviour pattern which is reflected by a leader in his role is known as the Leadership style. It is the result of the philosophy, personality, experience, and value system of a leader. The type of followers and the atmosphere prevailing in the organisation also affect leadership styles.

Types of Leadership Styles

1. Autocratic Leadership or Authoritative Leadership
2. Participative Leadership or Democratic Leadership
3. Free rein Leadership or Laissez-Faire Leadership

All the above-mentioned styles of leadership are used by a leader over a period of time. However, one style tends to predominate as his normal style of using power.

1 .Autocratic Leadership or Authoritative Leadership

The leadership style under which a leader centralises all decision-making powers and exercises full control over his subordinates is known as Autocratic or Authoritative leadership. The leader here gives orders and makes sure that they are obeyed. *For example,* if Sam assigns work and gives orders to complete the work as per his discretion without consulting his subordinates, then Sam is an Autocratic Leader.

- Policies and plans are made by the autocratic leader without consulting subordinates. The employees are also not given information about future plans.
- Orders are given and tasks are assigned, and subordinates are not given the freedom to influence the decision of the leader. This situation is similar to “bossing people around”.
- In such a leadership style, there is little or no concern for the welfare of the employees. Subordinates are compelled by the leaders to follow orders under the threat of penalties and punishments.
- Because of lack of freedom and threats of penalties and punishments, subordinates suffer from frustration and low morale. Subordinates avoid responsibility, lack initiative and become ‘Yes Men’.
- This leadership style should be used on rare occasions.
- Such leadership styles are used in the Military.
- This style is also known as the ‘Directive style of leadership’.

2 .Participative Leadership or Democratic Leadership

The leadership style under which a leader consults subordinates in the decision-making process and encourages them to give suggestions in setting goals and implementing decisions. In this, the subordinates are allowed to participate in the decision-making process and their suggestions are welcomed by the leader. *For example,* if Satyam is discussing the work and taking suggestions to complete the work, then he is a Participative or Democratic Leader.

- Under the Participative leadership style, orders are given only after consulting the subordinates, and any plan or policies is carried forwards only after the acceptance of the subordinates.
- This style wins greater confidence, cooperation, loyalty and initiative of the group. The morale of the employees is also boosted.
- Here, the subordinates are never asked to do things without working out long-term plans.
- The subordinates become part of the team and help the leader in making better decisions. Thus, it is a style of mutual benefit.

3 .Free rein Leadership or Laissez-Faire Leadership

The leadership style under which a leader gives complete freedom to the subordinates is known as Free rein or Laissez-Faire. *For example,* Sitaraman gives the responsibility of setting goals and devising plans to the subordinates, then here Sitaraman is practising free rein leadership.

- The leader depends on the group for setting goals and devising plans to achieve such goals.
- It can be said that this style is more of a non-leadership style than a leadership style, as the leader acts as an umpire and delegates the entire authority of decision-making to subordinates.
- Group members work as per their choice and competence. The leader acts as a contact man with the outsiders and collects information and resources required by the group to achieve the goals.
- This style is also known as “laissez-faire”, which means no interference from others. “Laissez-faire” is a french phrase, which means “let them do”.

What is the role of a project manager?

The Project Management Institute describes the role of the project manager as that of a change agent. They're someone who "makes project goals their own and uses their skills and expertise to inspire a sense of shared purpose within the project team."

Project managers are leaders — they not only ensure projects are delivered on time and within budget but must also engage and encourage their teams and inspire their clients. They need strong critical thinking capabilities to solve problems as they arise and finely-tuned communication skills (like a knack for customer service) to ensure everyone remains informed, motivated, and on board. It's no wonder that project managers are considered critical to the success of any venture.

Project manager responsibilities

A project manager is responsible for a variety of tasks and activities. The following list includes the key responsibilities every project manager takes on.

1. Planning

A project manager is responsible for formulating a project plan to meet the project's objectives while adhering to an approved budget and timeline. This blueprint will guide the project from ideation to fruition. It will include the project's scope, the resources necessary, the anticipated time and financial requirements, the communication strategy, a plan for execution and documentation, and a proposal for follow-up and maintenance. If the project has not yet gained approval, this plan will serve as a critical part of the pitch to key decision-makers.

2. Leading

An essential part of any project manager's role is to assemble and lead the project team. This requires excellent communication, people, and leadership skills, as well as a keen eye for others' strengths and weaknesses. Once the team has been created, the project manager assigns tasks, sets deadlines, provides necessary resources, and meets regularly with the members. An ability to speak openly and frequently with all stakeholders is critical.

3. Execution

The project manager participates in and supervises the successful execution of each stage of the project. Again, this requires frequent, open communication with the project team members and stakeholders.

4. Time management

Staying on schedule is crucial to completing any project, and time management is one of the key responsibilities of a project manager. Project managers are responsible for resolving derailments and communicating effectively with team members and other stakeholders to ensure the project

gets back on track. Project managers should be experts at risk management and contingency planning to continue moving forward even when roadblocks occur.

5. Budget

Project managers devise a budget for a project and stick to it as closely as possible. If certain parts of the project end up costing more (or, in a perfect world, less) than anticipated, project managers moderate the spend and reallocate funds when necessary.

6. Documentation

A project manager must develop effective ways to measure and analyze the project's progress. Common strategies for documenting a project include data collection and verbal and written status reports. It's also a project manager's job to ensure that all relevant actions are approved and that these documents are archived for future reference.

7. Maintenance

The work doesn't end once a project has been completed. There needs to be a plan for ongoing maintenance and troubleshooting. The project manager devises methods for properly supporting the final deliverable going forward, even if they are not directly overseeing its day-to-day operations.

B) Systems View of Project Management

A Systems View of Project Management refers to the perspective that considers a project as an integrated and interconnected system rather than a collection of isolated tasks. This approach recognizes that various components within a project are interdependent and that changes or developments in one area can impact the entire system. Here are some key points to consider in a systems view of project management:

1. Interconnected Components:
   - Projects are seen as a set of interconnected components, including people, processes, resources, technology, and external factors.
   - Changes in one component can have cascading effects on others, influencing the overall project performance.

2. Holistic Approach:
   - A systems view takes a holistic approach, considering the project as a whole rather than focusing solely on individual tasks or phases.
   - It emphasizes understanding the relationships and dependencies between different elements of the project.

3. Feedback Loops:
   - Recognizes the presence of feedback loops, where outputs from one part of the project can affect inputs or processes in another.
   - This perspective helps in identifying and managing potential issues before they escalate.

4. Adaptability and Flexibility:
   - Acknowledges the need for adaptability and flexibility as projects evolve and encounter uncertainties.
   - The ability to respond to changes in one part of the system without disrupting the entire project is a crucial aspect.

5. Integration of Project and Organizational Systems:
   - Considers the integration of the project system with the larger organizational system.
   - Aligning project goals and outcomes with the strategic objectives of the organization is essential for overall success.

6. Life Cycle Perspective:
   - Takes into account the entire life cycle of the project, from initiation to closure.
   - Understanding how each phase contributes to the overall success and how transitions between phases are managed is crucial.

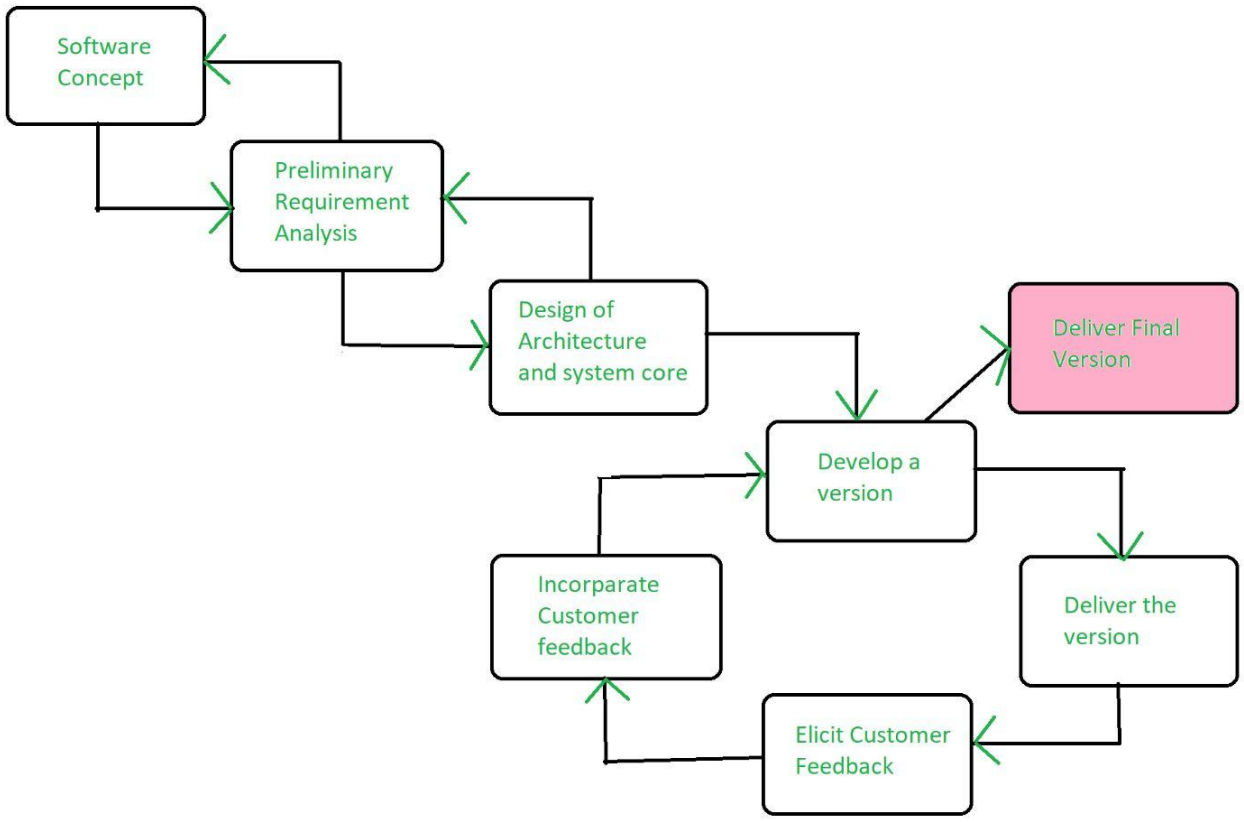
7. Risk Management:
   - Incorporates risk management as an integral part of the system view, recognizing that risks can emerge at any stage and impact various aspects of the project.

8. Stakeholder Involvement:
   - Recognizes the importance of stakeholders and their influence on the project system.
   - Communication and collaboration with stakeholders are essential to ensure that their needs and expectations are considered throughout the project life cycle.

In summary, adopting a systems view in project management allows for a more comprehensive understanding of the project's dynamics and enhances the ability to navigate complexities by considering the interdependencies and interactions among various elements. This approach supports effective decision-making, risk management, and overall project success.

Q . 2 Evolutionary model

Evolutionary model is a combination of Iterative and Incremental model of software development life cycle. Delivering your system in a big bang release, delivering it in incremental process over time is the action done in this model. Some initial requirements and architecture envisioning need to be done. It is better for software products that have their feature sets redefined during development because of user feedback and other factors. The Evolutionary development model divides the development cycle into smaller, incremental waterfall models in which users are able to get access to the product at the end of each cycle. Feedback is provided by the users on the product for the planning stage of the next cycle and the development team responds, often by changing the product, plan or process. Therefore, the software product evolves with time. All the models have the disadvantage that the duration of time from start of the project to the delivery time of a solution is very high. Evolutionary model solves this problem in a different approach.



Evolutionary model suggests breaking down of work into smaller chunks, prioritizing them and then delivering those chunks to the customer one by one. The number of chunks is huge and is the number of deliveries made to the customer. The main advantage is that the customer's confidence increases as he constantly gets quantifiable goods or services from the beginning of the project to verify and validate his requirements. The model allows for changing requirements as well as all work is broken down into maintainable work chunks.

Application of Evolutionary Model:

1. It is used in large projects where you can easily find modules for incremental implementation. Evolutionary model is commonly used when the customer wants to start using the core features instead of waiting for the full software.
2. Evolutionary model is also used in object oriented software development because the system can be easily portioned into units in terms of objects.

Necessary conditions for implementing this model:-

- Customer needs are clear and been explained in deep to the developer team.
- There might be small changes required in separate parts but not a major change.
- As it requires time, so there must be some time left for the market constraints.
- Risk is high and continuous targets to achieve and report to customer repeatedly.
- It is used when working on a technology is new and requires time to learn.

Advantages:

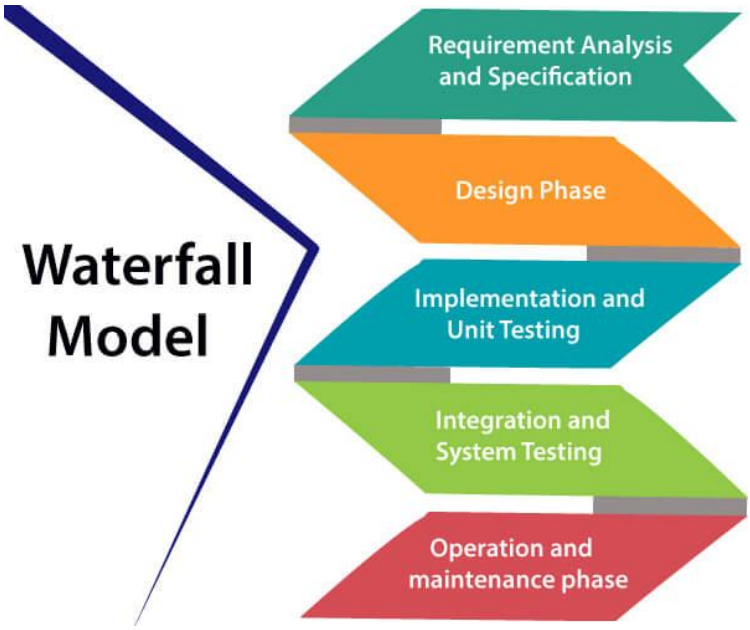
- In evolutionary model, a user gets a chance to experiment partially developed system.
- It reduces the error because the core modules get tested thoroughly.

Disadvantages:

- Sometimes it is hard to divide the problem into several versions that would be acceptable to the customer which can be incrementally implemented and delivered.

Q .3 WaterFall Model

1. Requirements analysis and specification phase: The aim of this phase is to understand the exact requirements of the customer and to document them properly. Both the customer and the software developer work together so as to document all the functions, performance, and interfacing requirement of the software. It describes the "what" of the system to be produced and not "how."In this phase, a large document called Software Requirement Specification (SRS) document is created which contained a detailed description of what the system will do in the common language.



2. Design Phase: This phase aims to transform the requirements gathered in the SRS into a suitable form which permits further coding in a programming language. It defines the overall software architecture together with high level and detailed design. All this work is documented as a Software Design Document (SDD).

3. Implementation and unit testing: During this phase, design is implemented. If the SDD is complete, the implementation or coding phase proceeds smoothly, because all the information needed by software developers is contained in the SDD.

During testing, the code is thoroughly examined and modified. Small modules are tested in isolation initially. After that these modules are tested by writing some overhead code to check the interaction between these modules and the flow of intermediate output.

4. Integration and System Testing: This phase is highly crucial as the quality of the end product is determined by the effectiveness of the testing carried out. The better output will lead to satisfied customers, lower maintenance costs, and accurate results. Unit testing determines the efficiency of individual modules. However, in this phase, the modules are tested for their interactions with each other and with the system.

5. Operation and maintenance phase: Maintenance is the task performed by every user once the software has been delivered to the customer, installed, and operational.

Advantages of Waterfall model

- **This model is simple to implement also the number of resources that are required for it is minimal.**
- **The requirements are simple and explicitly declared; they remain unchanged during the entire project development.**
- **The start and end points for each phase is fixed, which makes it easy to cover progress.**
- **The release date for the complete product, as well as its final cost, can be determined before development.**
- **It gives easy to control and clarity for the customer due to a strict reporting system.**

Disadvantages of Waterfall model

- **In this model, the risk factor is higher, so this model is not suitable for more significant and complex projects.**
- **This model cannot accept the changes in requirements during development.**
- **It becomes tough to go back to the phase. For example, if the application has now shifted to the coding phase, and there is a change in requirement, It becomes tough to go back and change it.**

D) Agile Model .

The meaning of Agile is swift or versatile."Agile process model" refers to a software development approach based on iterative development. Agile methods break tasks into smaller iterations, or parts do not directly involve long term planning. The project scope and requirements are laid down at the beginning of the development process. Plans regarding the number of iterations, the duration and the scope of each iteration are clearly defined in advance.

Each iteration is considered as a short time "frame" in the Agile process model, which typically lasts from one to four weeks. The division of the entire project into smaller parts helps to minimize the project risk and to reduce the overall project delivery time requirements. Each iteration involves a team working through a full software development life cycle including planning, requirements analysis, design, coding, and testing before a working product is demonstrated to the client.

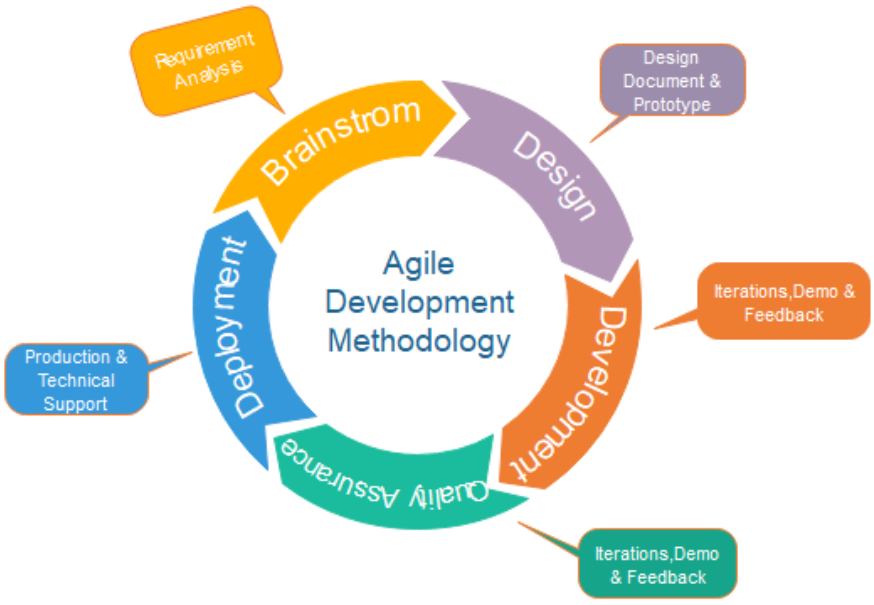


Fig. Agile Model

Phases of Agile Model:

Following are the phases in the Agile model are as follows:

1. **Requirements gathering**
2. **Design the requirements**
3. **Construction/ iteration**
4. **Testing/ Quality assurance**
5. **Deployment**
6. **Feedback**

1. Requirements gathering: In this phase, you must define the requirements. You should explain business opportunities and plan the time and effort needed to build the project. Based on this information, you can evaluate technical and economic feasibility.

2. Design the requirements: When you have identified the project, work with stakeholders to define requirements. You can use the user flow diagram or the high-level UML diagram to show the work of new features and show how it will apply to your existing system.

3. Construction/ iteration: When the team defines the requirements, the work begins. Designers and developers start working on their project, which aims to deploy a working product. The product will undergo various stages of improvement, so it includes simple, minimal functionality.

4. Testing: In this phase, the Quality Assurance team examines the product's performance and looks for the bug.

5. Deployment: In this phase, the team issues a product for the user's work environment.

6. Feedback: After releasing the product, the last step is feedback. In this, the team receives feedback about the product and works through the feedback.

Agile Testing Methods:

- **Scrum**
- **Crystal**
- **Dynamic Software Development Method(DSDM)**
- **Feature Driven Development(FDD)**
- **Lean Software Development**
- **eXtreme Programming(XP)**

Scrum

SCRUM is an agile development process focused primarily on ways to manage tasks in team-based development conditions.

There are three roles in it, and their responsibilities are:

- **Scrum Master: The scrum can set up the master team, arrange the meeting and remove obstacles for the process**
- **Product owner: The product owner makes the product backlog, prioritizes the delay and is responsible for the distribution of functionality on each repetition.**
- **Scrum Team: The team manages its work and organizes the work to complete the sprint or cycle.**

eXtreme Programming(XP)

This type of methodology is used when customers are constantly changing demands or requirements, or when they are not sure about the system's performance.

Crystal:

There are three concepts of this method-

1. **Chartering: Multi activities are involved in this phase such as making a development team, performing feasibility analysis, developing plans, etc.**
2. **Cyclic delivery: under this, two more cycles consist, these are:**
    A. **Team updates the release plan.**
    B. **Integrated product delivers to the users.**
3. **Wrap up: According to the user environment, this phase performs deployment, post-deployment.**

Dynamic Software Development Method(DSDM):

DSDM is a rapid application development strategy for software development and gives an agile project distribution structure. The essential features of DSDM are that users must be actively connected, and teams have been given the right to make decisions. The techniques used in DSDM are:

1. **Time Boxing**
2. **MoSCoW Rules**
3. **Prototyping**

The DSDM project contains seven stages:

1. **Pre-project**
2. **Feasibility Study**
3. **Business Study**
4. **Functional Model Iteration**
5. **Design and build Iteration**
6. **Implementation**
7. **Post-project**

Feature Driven Development(FDD):

This method focuses on "Designing and Building" features. In contrast to other smart methods, FDD describes the small steps of the work that should be obtained separately per function.

Lean Software Development:

Lean software development methodology follows the principle "just in time production." The lean method indicates the increasing speed of software development and reducing costs. Lean development can be summarized in seven phases.

1. **Eliminating Waste**
2. **Amplifying learning**
3. **Defer commitment (deciding as late as possible)**
4. **Early delivery**
5. **Empowering the team**
6. **Building Integrity**
7. **Optimize the whole**

When to use the Agile Model?

- **When frequent changes are required.**
- **When a highly qualified and experienced team is available.**
- **When a customer is ready to have a meeting with a software team all the time.**
- **When project size is small.**

Advantage(Pros) of Agile Method:

1. **Frequent Delivery**
2. **Face-to-Face Communication with clients.**
3. **Efficient design and fulfils the business requirement.**
4. **Anytime changes are acceptable.**
5. **It reduces total development time.**

Disadvantages(Cons) of Agile Model:

1. **Due to the shortage of formal documents, it creates confusion and crucial decisions taken throughout various phases can be misinterpreted at any time by different team members.**
2. **Due to the lack of proper documentation, once the project completes and the developers allotted to another project, maintenance of the finished project can become a difficulty.**

E) DFD

Characteristics of DFD

- DFDs are commonly used during problem analysis.
- DFDs are quite general and are not limited to problem analysis for software requirements specification.
- DFDs are very useful in understanding a system and can be effectively used during analysis.
- It views a system as a function that transforms the inputs into desired outputs.
- The DFD aims to capture the transformations that take place within a system to the input data so that eventually the output data is produced.
- The processes are shown by named circles and data flows are represented by named arrows entering or leaving the bubbles.
- A rectangle represents a source or sink and it is a net originator or consumer of data. A source sink is typically outside the main system of study.

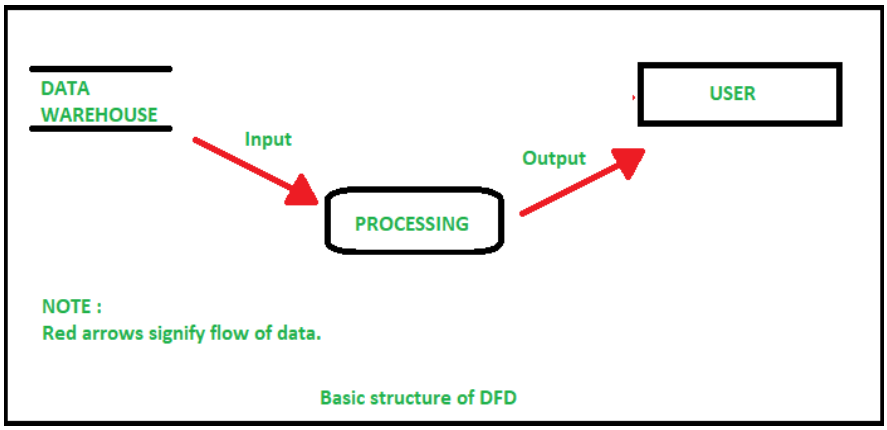
Components of DFD

The Data Flow Diagram has 4 components:

- Process Input to output transformation in a system takes place because of process function. The symbols of a process are rectangular with rounded corners, oval, rectangle or a circle. The process is named a short sentence, in one word or a phrase to express its essence
- Data Flow Data flow describes the information transferring between different parts of the systems. The arrow symbol is the symbol of data flow. A relatable name should be given to the flow to determine the information which is being moved. Data flow also represents material along with information that is being moved. Material shifts are modeled in systems that are not merely informative. A given flow should only transfer a single type of information. The direction of flow is represented by the arrow which can also be bi-directional.
- Warehouse The data is stored in the warehouse for later use. Two horizontal lines represent the symbol of the store. The warehouse is

simply not restricted to being a data file rather it can be anything like a folder with documents, an optical disc, a filing cabinet. The data warehouse can be viewed independent of its implementation. When the data flow from the warehouse it is considered as data reading and when data flows to the warehouse it is called data entry or data updating.

- Terminator The Terminator is an external entity that stands outside of the system and communicates with the system. It can be, for example, organizations like banks, groups of people like customers or different departments of the same organization, which is not a part of the model system and is an external entity. Modeled systems also communicate with terminator.



Basic structure of DFD

Rules for creating DFD

- The name of the entity should be easy and understandable without any extra assistance(like comments).
- The processes should be numbered or put in ordered list to be referred easily.
- The DFD should maintain consistency across all the DFD levels.
- A single DFD can have a maximum of nine processes and a minimum of three processes.

Symbols Used in DFD

- Square Box: A square box defines source or destination of the system. It is also called entity. It is represented by rectangle.

- **Arrow or Line: An arrow identifies the data flow i.e. it gives information to the data that is in motion.**
- **Circle or bubble chart: It represents as a process that gives us information. It is also called processing box.**
- **Open Rectangle: An open rectangle is a data store. In this data is store either temporary or permanently.**

Levels of DFD

DFD uses hierarchy to maintain transparency thus multilevel DFD's can be created. Levels of DFD are as follows:

- **0-level DFD: It represents the entire system as a single bubble and provides an overall picture of the system.**
- **1-level DFD: It represents the main functions of the system and how they interact with each other.**
- **2-level DFD: It represents the processes within each function of the system and how they interact with each other.**
- **3-level DFD: It represents the data flow within each process and how the data is transformed and stored.**

Advantages of DFD

- **It helps us to understand the functioning and the limits of a system.**
- **It is a graphical representation which is very easy to understand as it helps visualize contents.**
- **Data Flow Diagram represent detailed and well explained diagram of system components.**
- **It is used as the part of system documentation file.**
- **Data Flow Diagrams can be understood by both technical or nontechnical person because they are very easy to understand.**

Disadvantages of DFD

- **At times DFD can confuse the programmers regarding the system.**
- **Data Flow Diagram takes long time to be generated, and many times due to this reasons analysts are denied permission to work on it.**

F)Change Management

Change Management in software development refers to the transition from an existing state of the software product to another improved state of the product. It controls, supports, and manages changes to artifacts, such as code changes, process changes, or documentation changes. Where CCP (Change Control Process) mainly identifies, documents, and authorizes changes to a software application.

Each software development process follows Software Development Life Cycle (SDLC) where each phase is accordingly followed to finally deliver a good quality software product. Change Management does not come under any phases of SDLC still it has great importance in the entire software development process. There are various types of change management tools that are used for various purposes like to adopt, control, represent and effect the change required. For example Change management tools for Flow Charting, Project Planning, Data collection, etc.

Process of Change Management :

When any software application/product goes for any changes in an IT environment, it undergoes a series of sequential processes as follows:

- Creating a request for change
- Reviewing and assessing a request for change
- Planning the change
- Testing the change
- Creating a change proposal
- Implementing changes
- Reviewing change performance
- Closing the process

Importance of Change Management :

- For improving performance
- For increasing engagement
- For enhancing innovation
- For including new technologies
- For implementing new requirements

- For reducing cost

Source of Change :

There may be multiple reasons involved during the development process for which certain changes are required to be implemented in the product. These sources are as follows :

- Business reorganization
- New Market conditions
- New equipment
- Fixing any bugs/errors
- New customer needs
- Performance or reliability improvement
- Budgetary or scheduling constraints

Key points to be considered during Change Management :

- Reason of change
- Result of change
- The portion to be changed
- Person will change
- Risks involved in change
- Alternative to change
- Resources required for change
- Relationship between changes

G)PERT

Project Evaluation and Review Technique (PERT) is a procedure through which activities of a project are represented in its appropriate sequence and timing. It is a scheduling technique used to schedule, organize and integrate tasks within a project. PERT is basically a mechanism for management planning and control which provides blueprint for a particular project. All of the primary elements or events of a project have been finally identified by the PERT. In this technique, a PERT Chart is made which represent a schedule for all the specified tasks in the project. The reporting levels of the tasks or events in the PERT Charts is somewhat same as defined in the work breakdown structure (WBS).

The PERT chart is used to schedule, organize and co-ordinate tasks within the project. the objective of PERT chart is to determine the critical path, which comprises critical activities that should be completed on schedule. This chart is prepared with the help of information generated in project planning activities such as estimation of effort, selection of suitable process model for software development and decomposition of tasks into subtasks.

Characteristics of PERT:

The main characteristics of PERT are as following :

1. **It serves as a base for obtaining the important facts for implementing the decision-making.**
2. **It forms the basis for all the planning activities.**
3. **PERT helps management in deciding the best possible resource utilization method.**
4. **PERT take advantage by using time network analysis technique.**
5. **PERT presents the structure for reporting information.**
6. **It helps the management in identifying the essential elements for the completion of the project within time.**
7. **It specifies the activities that from the critical path.**
8. **It describes the probability of completion of project before the specified date.**
9. **It describes the dependencies of one or more tasks on each other.**
10. **It represents the project in graphical plan form.**

Advantages of PERT:

It has the following advantages :

1. **Estimation of completion time of project is given by the PERT.**
2. **It supports the identification of the activities with slack time.**
3. **The start and dates of the activities of a specific project is determined.**
4. **It helps project manager in identifying the critical path activities.**
5. **PERT makes well organized diagram for the representation of large amount of data.**

Disadvantages of PERT:

It has the following disadvantages :

1. **The complexity of PERT is more which leads to the problem in implementation.**
2. **The estimation of activity time are subjective in PERT which is a major disadvantage.**
3. **Maintenance of PERT is also expensive and complex.**
4. **The actual distribution of may be different from the PERT beta distribution which causes wrong assumptions.**