

ASSIGNMENT NUMBER- 01

ASDM

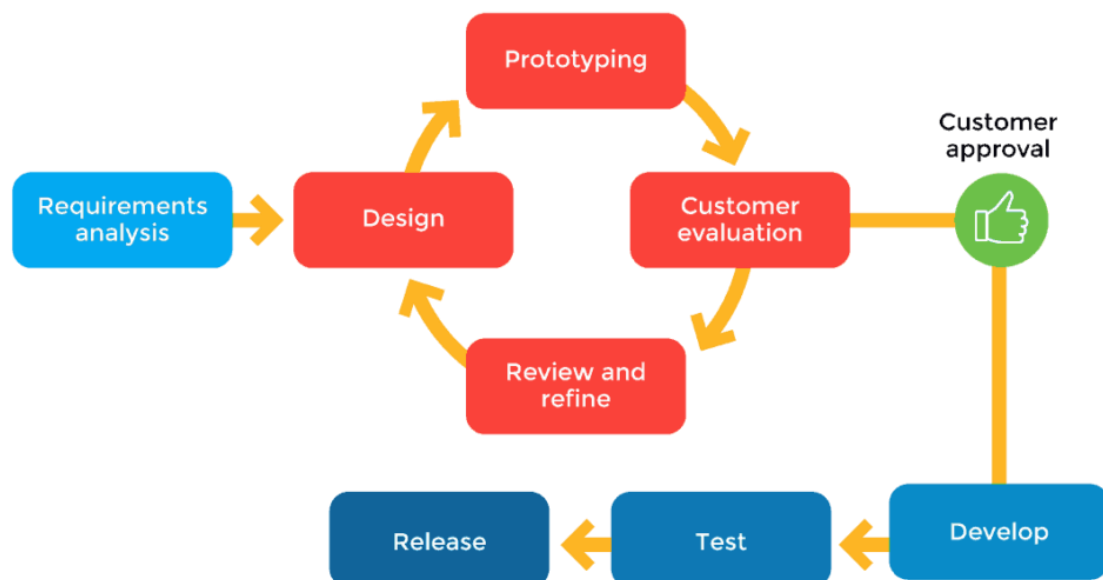
Rishabh Tyagi

Form Number-: 220703202

Q.1 Discuss the prototyping model. What is the effect of designing a prototype on the overall cost of the project?

Solution-:

Prototyping is defined as the process of developing a working replication of a product or system that has to be engineered. It offers a small scale facsimile of the end product and is used for obtaining customer feedback as described below:



The Prototyping Model is one of the most popularly used Software Development Life Cycle Models (SDLC models). This model is used when the customers do not know the exact project requirements beforehand. In this model, a prototype of the end product is first developed, tested and refined as per customer feedback repeatedly till a final acceptable prototype is achieved which forms the basis for developing the final product.

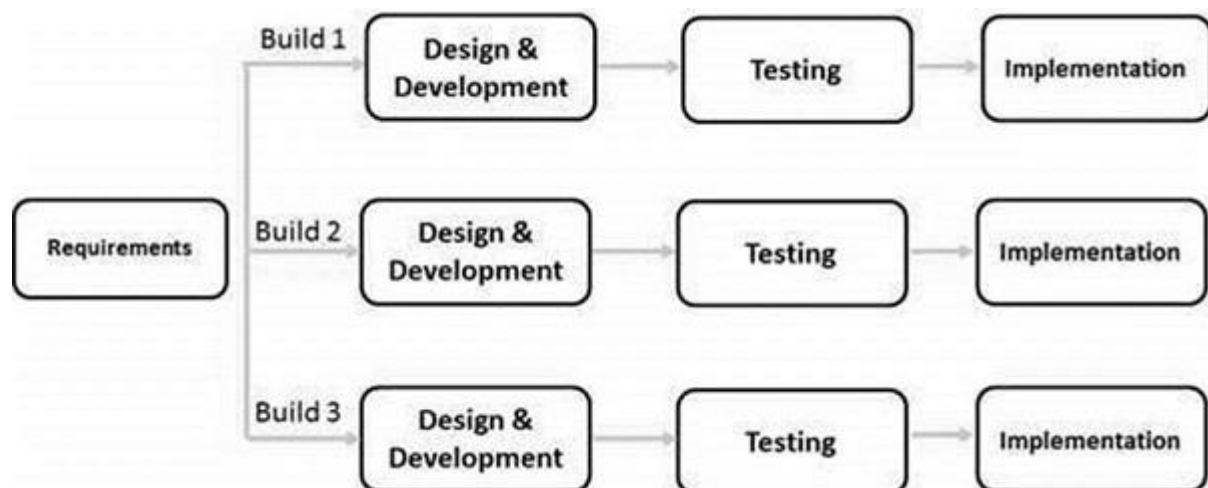
In this process model, the system is partially implemented before or during the analysis phase thereby giving the customers an opportunity to see the product early in the life cycle. The process starts by interviewing the customers and developing the incomplete high-level paper model. This document is used to build the initial prototype supporting only the basic functionality as desired by the customer. Once the customer figures out the problems, the prototype is further refined to eliminate them. The process continues until the user approves the prototype and finds the working model to be satisfactory.

Prototyping may have some initial costs of developing, but it reduces the overall budget by helping your product to be free of the errors or glitches that could have occurred if the idea was made from scratch without any prior user testing. Furthermore, prototyping also helps to understand the intrinsic flaws, shortcomings and drawbacks that can be improved during the product development process. If the prototyping process is ignored completely, it might result in the restructuring and redesigning of the entire product after spending all your resources on its development. So, the effect of designing a prototype on the overall cost of a software project is to actually reduce the additional costs of restructuring and reframing it after its full-fledged development- which might cost a fortune.

Q.2 Compare iterative enhancement model and evolutionary process model.

Solution-:

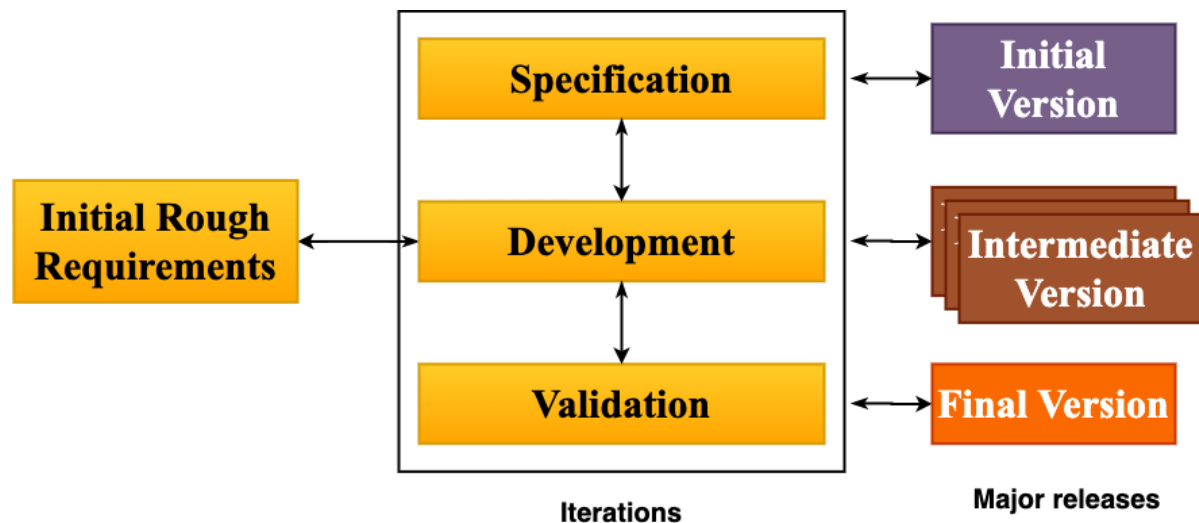
(A.) Iterative enhancement model -: The iterative enhancement model in software engineering combines elements of the linear sequential model with the iterative philosophy of prototyping. In this model, the software is broken down into several modules which are incrementally developed and delivered. Firstly, the development team develops the core module of the system. After that, it is refined into increasing levels of capacity of adding new functionalities in successive versions.



When it is used the first increment is often a core product i.e. basic requirements are addressed but many supplementary features are undelivered. The core product is used by the customers. As a result of use or evaluation, a plan is developed for the next increment. The plan addresses the modification of the core product to meet the needs of customers, and the delivery of additional features and functionality. This process is repeated following each delivery of increment until the complete product is produced.

(B.) Evolutionary process model-: In the evolutionary model, the core modules of the software are first developed, which is refined into an increasing level of capabilities called iterations. In iteration, new functionalities will be added, and also the existing functionalities can change based on the user feedback. Each iteration is developed through a mini waterfall

model, and at the end of an iteration, a tested, integrated executable system is developed and deployed at the customer site. This is also called a minor release of the software product.



At the end of development, maybe the existing functionalities might have changed. No new functionalities might have got added. Still, only the existing functionalities have been refined, or it may so happen that existing functionalities have changed a little bit, and new functionalities have been added. So, these are an increment, and therefore, it has some incremental development. This is called a major release of the software product. Before a major release, there would be multiple minor releases of the software product.

Q.3 As we move outward along with process flow path of the spiral model, what can we say about software that is being developed or maintained.

Solution-:

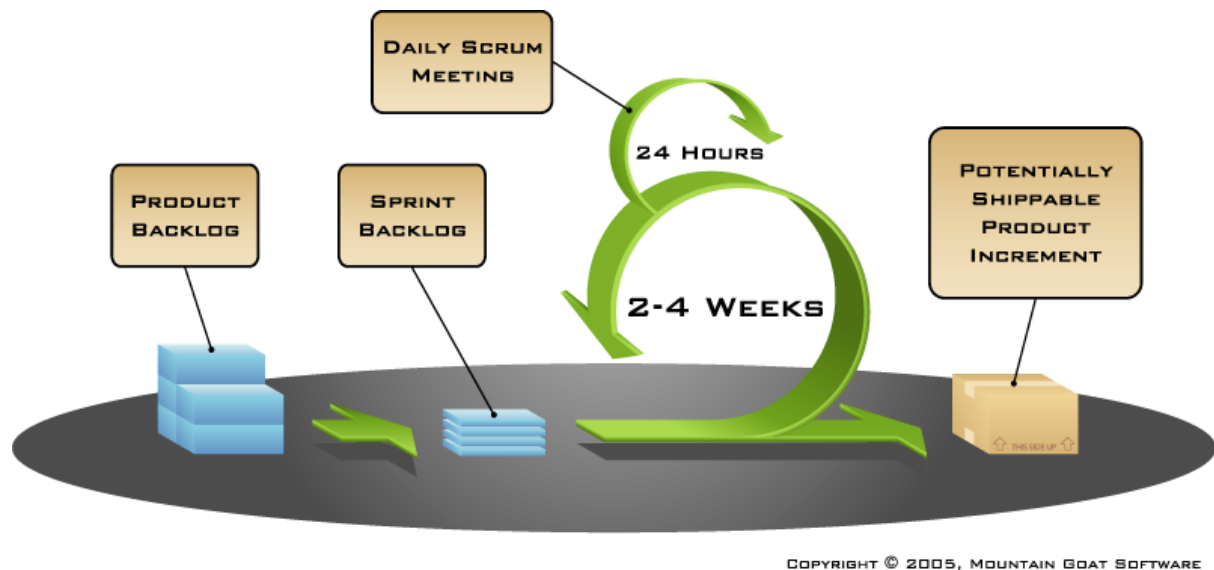
The spiral model, initially proposed by Boehm, is an evolutionary software process model that couples the iterative feature of prototyping with the controlled and systematic aspects of the linear sequential model. It implements the potential for rapid development of new versions of the software. Using the spiral model, the software is developed in a series of incremental releases. During the early iterations, the additional release may be a paper model or prototype. During later iterations, more and more complete versions of the engineered system are produced. As work moves outward on the spiral, the product moves toward a more complete state and the level of abstraction at which work is performed is reduced (i.e., implementation specific work accelerates as we move further from the origin).

Q.4 Explain the Scrum Agile methodology.

Solution-:

Scrum is a methodology for projects. It falls under the agile methodology and defines roles, procedures, tools, processes to make sure to deliver an efficient and effective

project well on time through iterative development cycles. As per a report, there are almost 70% of the software teams who use scrum or scrum hybrid.



This methodology is basically followed where there is the demand of high development process, high involvement of stakeholders. Scrum methodology repeatedly monitors software development while the project is being developed.

Scrum Software Development Methodology has a major focus on the responsibility, teamwork, and iterative progress towards a well-defined business goal.

Here are the main roles involved in the development process, according to the Scrum model:

- The product owner takes care of the end user's interests;
- The Scrum master coordinates the whole development process. Another task is to make sure that Scrum is used properly and to hold regular Scrum meetings;
- The Scrum team develops the product. Its main tasks are programming, analysis, testing, etc.

Phases of Scrum Model-:

Step 1. Product Backlog Creation

Step 2. Sprint Planning and Sprint Backlog Creation

Step 3. Working on the Sprint. Daily Scrum Meetings

Step 4. Product Increment and Sprint Review

Step 5. Retrospective and Next Sprint Planning

Advantages:

- Transparent system pushes developers to comply with their assignments and deliver it on time
- Defined deadline at every step keep developers motivated and empowered at every step
- Feedback at every level of the project ensures that quality project is delivered in the end

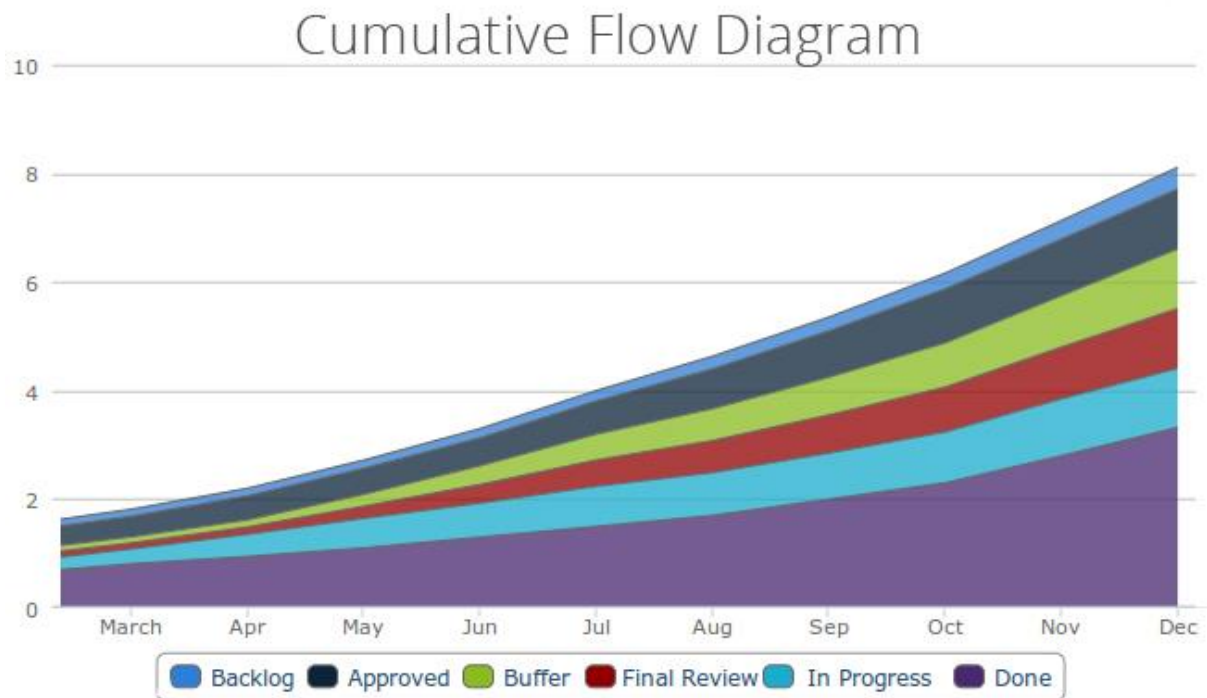
Disadvantages:

- Difficult to plan, structure and organize a project with no clear mission and vision
- Frequent changes in the project lead to a delay in the delivery time of the project
- Utilizes more resources and stakeholder's involvement in every small detail change and discussion.

Q.5 Explain the utility of Kanban CFD reports.

Solution-: A cumulative flow diagram (CFD) is an advanced analytic tool in the Kanban method. It provides teams with a visualization of workflow efforts and overall project progress. The cumulative flow diagram allows teams to monitor how stable their workflow is, anticipate bottlenecks so they can adjust their workflow accordingly, and help make processes more predictable.

By graphing how tasks accumulate over time and their overall distribution across the process stages, cumulative flow diagrams visualize massive amounts of data from which you can gather quantitative and qualitative insights into past and present problems with your workflow stability—and where you should focus on making your processes more efficient.



In the Kanban methodology, Kanban boards are used to divide the workflow of a given project into three columns: “To Do” tasks, tasks that are “Work in Progress” (WIP), and tasks that are “Done.” Cumulative flow diagrams collect every task that has gone through your workflow to visualize three critical metrics:

- **Cycle time:** This is the total time it takes your team to complete each task from the beginning to the end. One of the benefits of CFDs is that you can see where you can optimize your workflow to reduce cycle times.
- **Work in progress:** This is the number of tasks your team is actively handling at a certain time. Cumulative flow diagrams will visualize inefficiencies in your project timeline when your team has too much or too little work in progress at any given point.
- **Throughput:** This is the number of tasks your team can complete over a given period. As this is the ultimate measure of your team’s productivity, cumulative flow diagrams should show where you can align your efforts and resources so that throughput increases over time.