Q1: Discuss the prototyping model. What is the effect of designing a prototype on the overall cost of the project?

The prototyping model is a systems development method in which a prototype is built, tested and then reworked as necessary until an acceptable outcome is achieved from which the complete system or product can be developed.

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.

Q2: Compare iterative enhancement model and evolutionary process model.

Iterative Enhancement Model: This model has the similar phases as the waterfall model, but with fewer restrictions. In general the phases occur in the same order as in the waterfall model but these may be conducted in several cycles. A utilizable product is released at the end of the each cycle with each release providing additional functionality.

Evolutionary Development Model: Evolutionary development model bear a resemblance to iterative enhancement model. The similar phases as defined for the waterfall model occur here in a cyclical fashion. This model is different from iterative enhancement model in the sense that this doesn't require a useable product at the end of each cycle. In evolutionary development requirements are implemented by category rather than by priority.

Q3 : 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.

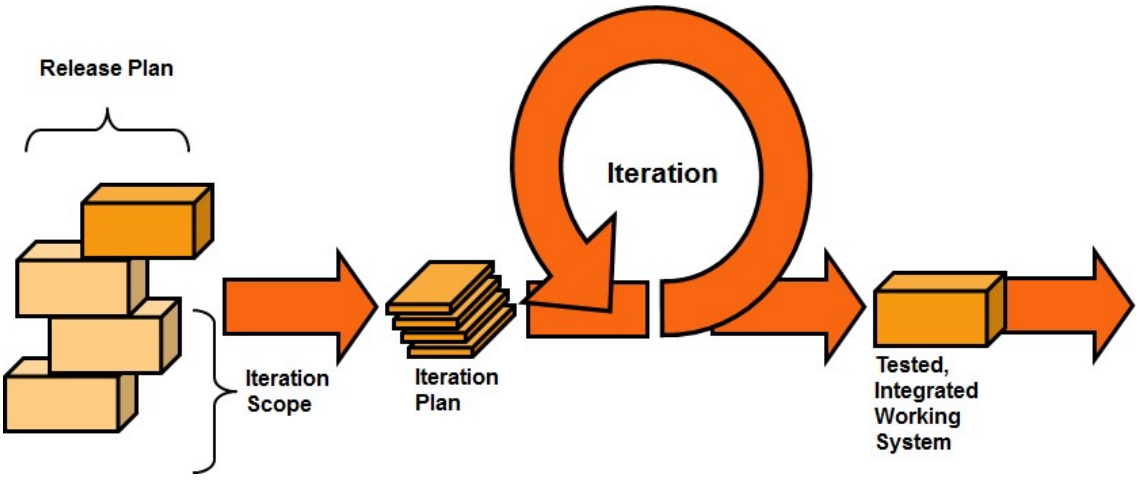
As work moves outward on the spiral the product moves toward a more complete state and the level of abstraction at which work is performed is reduced (implementation specific work accelerates as we move further from the origin).

Q4. Explain the Scrum Agile methodology.

Agile scrum methodology is a project management system that relies on incremental development. Each iteration consists of two- to four-week sprints, where the goal of each sprint is to build the most important features first and come out with a Potentially Shippable Product.

One of the most popular agile methodologies in use today, Scrum is a lightweight software development methodology that focuses on having **small time-boxed sprints** of new functionality that are incorporated into an **integrated product baseline**. Scrum places an emphasis on transparent customer interaction, feedback and adjustments rather than documentation and prediction.

Instead of phases, Scrum projects are broken down into releases and sprints. At the end of each sprint you have a fully functioning system that could be released:



Q5: Explain the utility of Kanban CFD reports.

CFD charts are a powerful tool that Kanban teams can use to **measure flow and analyze trends about a team's performance**. Think of a CFD chart as a storyteller. It paints a picture of how workflows through your Kanban system within a period.

Cumulative Flow Diagram is an analytical tool, fundamental to Kanban method. It allows teams to visualize their effort and project progress. When there's an impediment about to occur within the process - the CFD is where you'll see it first. Instead of the graph staying smooth and rising gently, there will be a bump, a sudden ascend or descend. So, where being able to predict problems is concerned, this is the very graph you need.