

A18 -

1. Discorra sobre os conceitos de Reuso, Frameworks, Componentes e a diferença entre Reuso de Componentes e Frameworks.

R- A **reutilização de software** se baseia no uso de conceitos, produtos ou soluções previamente elaboradas ou adquiridas para criação de um novo software, visando melhorar significativamente a qualidade e a produtividade. Reusar um produto significa poder reusar partes de um sistema desenvolvido anteriormente como: especificações, módulos de um projeto, arquitetura e código fonte. A principal motivação para a reutilização está relacionada ao aumento dos níveis de qualidade e produtividade no desenvolvimento de software.

O **framework** nada mais é do que uma ferramenta que vai te ajudar a ter como único objetivo focar em desenvolver o projeto, não em detalhes de configurações. Sendo assim, o framework trouxe a prática de evitar que tenhamos que fazer tarefas repetitivas, automatizando parte do trabalho. Pensando numa situação de desenvolvimento, se precisarmos criar um formulário de cadastro de usuário, ele sempre vai requerer algum tipo de validação como e-mail e senha. O framework já terá essa validação pronta para ser utilizada.

Basicamente **componentes de software** é um pedaço de software independente, mas que pode ser usado por outros softwares. Se assemelha muito a uma biblioteca, ou seja, uma pessoa cria um componente para acessar um banco de dados. Ao invés de eu criar uma classe ou um código para fazer o acesso ao banco de dados, eu incluo esse componente com o código já criado para fazer acesso ao meu banco de dados. É um subconjunto de um software porque pode ser reutilizado, pode estar contido em um software, mas independente porque funciona também sozinho/ separado de outros sistemas.

A diferença entre **framework e reuso de componentes** está no conceito das duas coisas, enquanto o framework é um código de um software inacabado que para ser finalizado precisará de uma implementação e cuidado, os componentes reusáveis são “elementos” de um código já existente que, desde que bem documentados, podem adiantar muitos passos no desenvolvimento do projeto.

2. Discorra sobre o vídeo deste link:

https://www.youtube.com/watch?v=XfvQWnRgxG0&t=3s&ab_channel=MindMaster Descreva as principais etapas do Scrum e seus Atores. Mínimo 15 linhas.

R- O vídeo é bem explicativo e consegue trazer com clareza algumas características e quando e onde devo utilizar o SCRUM, no começo ele fala sobre o que se trata o SCRUM onde o autor afirma que se trata de um “framework simples para gerenciar projetos complexos”, logo em seguida o vídeo utiliza um gráfico que demonstra quando se deve utilizar a metodologia ágil.

Com base no que o vídeo aborda, ele demonstra que a metodologia ágil SCRUM, deve ser utilizada conforme os seguintes aspectos, quando tanto os requisitos, quanto a tecnologia que vai ser utilizada são desconhecidas, ou seja, é um cenário caótico que o SCRUM é melhor aplicado.

Em seguida, é abordado os três pilares do método ágil SCRUM, são eles: Transparência, Inspeção e Adaptação. A transparência, no SCRUM deve ser evidente, pois é necessário saber de tudo que se está acontecendo para que os processos ocorram da melhor forma possível. Já na inspeção, é verificado de forma constante o que está sendo feito em cada etapa da aplicação, ou seja, verificar o que já foi feito, o que está sendo feito e o que falta ser feito. E por fim, temos a adaptação que corresponde a maneira como o projeto se adapta conforme a sua demanda e crescimento ou diminuição de recursos.

Também foi possível aprender no vídeo sobre as práticas fundamentais do scrum. Em primeiro lugar vem os papéis básicos: Scrum Master (É o responsável por manter o SCRUM funcionando), Product Owner (líder em relação ao produto. Fica responsável pelas funcionalidades e recursos do projeto) e Dev Team (Auto organizada em termos de buscar o resultado esperado pelo product owner). Esses papéis são fundamentais para a metodologia. Em segundo, os eventos básicos: Planejar o Sprint, Executar, revisar, reuniões diárias e a retrospectiva do Sprint. Para finalizar no terceiro, os artefatos gerados que são o backlog do produto, o Sprint Backlog e a entrega ainda que incompleta do produto.

3. Discorra sobre Prova de Conceito (POC) e por que esta técnica tem relação com a Análise e Projeto de Software. Mínimo 10 linhas.

R- Prova de Conceito ou PoC, é uma palavra que vem do inglês na qual significa Proof of Concept. Ela busca pela evidência documentada para que um software venha a ser bem-sucedido.

Por se tratar de um dos primeiros métodos para testar e avaliar uma ideia conceitual para software, produto ou solução para uma empresa, a prova de conceito na prática é como um procedimento cujo o objetivo do mesmo é provar a viabilidade de um conceito e validá-lo em baixa escala. A prova de conceito normalmente caracteriza uma implementação resumida, simples e incompleta, que antecede o protótipo do projeto e assim busca validar o software ainda na sua etapa conceitual.

Na prova de conceito normalmente emprega-se vários testes, a fim de validar se o produto final é positivamente satisfatório ou não. Além disso, a Poc também busca testar possíveis processos e procedimentos a serem implantados na execução do desenvolvimento. Esse teste poderá se definir quais as melhores práticas entre etapas para aplicá-las ou não ao projeto final.

Tem relação direta com a análise e projeto de softwares, pois um software é um projeto que pode ser pequeno, mas pode ser enorme, o que impõe a necessidade de avaliação de sua viabilidade. Com base nisso, conseguimos obter redução de custos e de erros, e o mais importante que é a satisfação do cliente.