**SERVIÇO NACIONAL DE APRENDIZAGEM INDUSTRIAL – SENAI**

Juan Nachbar Medeiros

**ATIVIDADE COMPENSAÇÃO**

**CURSO TÉCNICO EM DESENVOLVIMENTO DE SISTEMAS**

**GARÇA/SP**

**2024**

1. O que é um modelo de processo de software e qual a sua importância no desenvolvimento de sistemas?

Os processos de desenvolvimento de software são fundamentais para guiar a criação de software e garantir sua qualidade, controle e gerenciamento eficiente. Entre os modelos mais conhecidos estão o Modelo em Cascata, que segue uma abordagem linear e sequencial, o Modelo em Espiral, que é iterativo e foca na análise de riscos, e o Desenvolvimento Ágil, que valoriza a flexibilidade e a comunicação constante com o cliente através de métodos como Scrum e Kanban. Outro modelo importante é o V-Model, que alinha testes e desenvolvimento, e o Desenvolvimento Orientado a Objetos, que é ideal para projetos complexos através da modelagem com objetos e classes. A escolha do processo adequado depende das características do projeto, e adotar um processo bem definido é crucial para assegurar a qualidade do software, otimizar a comunicação entre as equipes e partes interessadas, e garantir a documentação e manutenção adequadas ao longo do tempo.

1. Como os modelos de processos de software evoluíram ao longo do tempo? Quais foram os principais fatores que impulsionaram essa evolução?

Os modelos de processos de software evoluíram ao longo do tempo para atender às necessidades do mercado de desenvolvimento de sistemas. Alguns dos modelos de desenvolvimento de software e suas características são:

* **Modelo Waterfall**

Uma das primeiras metodologias de desenvolvimento de software, também conhecida como modelo cascata.

* **Metodologias Ágeis**

Surgiram no final dos anos 90 e início dos anos 2000, como resposta às limitações do modelo Waterfall.

* **DevOps**

Uma evolução das práticas ágeis, que enfatiza a colaboração entre as equipes de desenvolvimento e operações.

* **Modelos evolutivos**

Caracterizados por iterações, que permitem o desenvolvimento de várias versões do produto.

* **Prototipação**

Utilizada quando o desenvolvedor não tem certeza sobre a eficiência de um algoritmo, a adaptabilidade de um sistema operacional, ou a forma de interação entre o cliente e o sistema.

Os principais fatores que impulsionaram a Revolução Industrial foram vários e interconectados. A invenção e o aperfeiçoamento da máquina a vapor na Inglaterra, ao longo do final do século XVII e da década de 1760, desempenharam um papel crucial, pois essa tecnologia utilizava o vapor gerado pelo aquecimento de água com carvão para fornecer energia e movimentar máquinas. Além disso, a disponibilidade de mão de obra foi fundamental, já que havia uma grande quantidade de trabalhadores prontos para se empregar nas novas indústrias. O estímulo ao livre comércio também contribuiu significativamente, facilitando o comércio e a circulação de bens. Por fim, o capital disponível para investimento possibilitou o financiamento necessário para o desenvolvimento e expansão das indústrias. Esses fatores juntos criaram um ambiente propício para o avanço industrial.

1. Quais são as vantagens e desvantagens do modelo em cascata? Em quais tipos de projetos ele é mais adequado?

As vantagens são:

**Estrutura clara e organizada**

**Previsibilidade**

**Controle**

**Redução de riscos**

As desvantagens são:

**Falta de flexibilidade**

**Dificuldade em lidar com incertezas**

**Demora na entrega de valor**

**Falta de feedback**

O método de cascata é mais utilizado em projetos com requisitos bem definidos, com prazos e orçamentos fixos, que não pedem muita flexibilidade. Um exemplo de sua aplicação seria na construção de um prédio pois suas etapas são bem definidas e sequencias.

1. Explique os princípios e valores dos métodos ágeis. Quais são as principais diferenças entre Scrum e Kanban?

A ideia central dos métodos ágeis são buscar uma melhora no alinhamento das pessoas, colaboração, melhoria contínua e, principalmente, manter o foco na satisfação do cliente.

As principais diferenças entre as metodologias Scrum e Kanban são:

* **Estrutura**

O Scrum é uma metodologia estruturada que define papéis, cerimônias e artefatos, enquanto o Kanban é um método ágil que se concentra na visualização do fluxo de trabalho.

* **Fluxo de trabalho**

No Scrum, o trabalho é dividido em Sprints com compromissos bem definidos, enquanto no Kanban o fluxo de trabalho é visualizado em um quadro Kanban, permitindo ajustes contínuos.

* **Flexibilidade**

O Scrum é menos flexível devido à estrutura e cerimônias definidas, enquanto o Kanban é altamente flexível, se adaptando facilmente a diferentes contextos.

* **Entregas**

No Scrum, as entregas são feitas no final de cada sprint e são validadas pelo Product Owner, enquanto no Kanban a decisão de como fazer fica ao critério da equipe.

* **Priorização**

No Kanban, as tarefas priorizadas são concluídas primeiro, pois a equipe decide coletivamente o que é melhor.

* **Medição de entregas**

O Kanban utiliza o work in progress, o que limita o número de atividades, enquanto o Scrum foca na velocidade, ou seja, na produtividade alcançada em cada sprint.

1. Quais são os maiores desafios enfrentados na implementação de um modelo de processo de software em uma organização??

Os principais desafios e barreiras encontrados no desenvolvimento dos softwares são os: recursos financeiros restritos; custos de infraestrutura; necessidade de retrabalho; dificuldade de suporte com os fornecedores

1. Qual é a relação entre a formação acadêmica e a qualidade do software desenvolvido por profissionais de tecnologia da informação?

Um treinamento acadêmico sólido fornece uma base teórica robusta, ensina melhores práticas e metodologias, e estimula o desenvolvimento do pensamento crítico. Tudo isso eleva a qualidade do software, contribuindo para a criação de soluções criativas e eficazes.

1. Como a prática de revisão de código (code review) influencia a qualidade do software e o desenvolvimento das habilidades técnicas dos desenvolvedores?

Ela é essencial para aprimorar a qualidade do software, ajudando a identificar e corrigir defeitos antes da integração. Além disso, assegura a aplicação das melhores práticas e oferece aos desenvolvedores um feedback valioso sobre o trabalho em andamento.

1. Quais são os efeitos da documentação técnica bem elaborada na manutenção e escalabilidade de projetos de software?

Ela é essencial para aprimorar a qualidade do software, ajudando a identificar e corrigir defeitos antes da integração. Além disso, assegura a aplicação das melhores práticas e oferece aos desenvolvedores um feedback valioso sobre o trabalho em andamento.

REFERÊNCIAS

<https://www.dio.me/articles/os-processos-de-desenvolvimento-de-software-engenharia-de-software>

<https://www.devmedia.com.br/introducao-aos-processos-de-software-e-o-modelo-incremental-e-evolucionario/29839#:~:text=Os%20processos%20evolucion%C3%A1rios%20se%20caracterizam,que%20est%C3%A1%20para%20ser%20constru%C3%ADdo>.

<https://www.dio.me/articles/a-evolucao-das-metodologias-de-desenvolvimento-de-software-do-waterfall-ao-devops> <https://mundoeducacao.uol.com.br/historiageral/revolucao-industrial-2.htm#:~:text=Antecedentes%20da%20Revolu%C3%A7%C3%A3o%20Industrial,foi%20aprimorado%20por%20James%20Watt>.

<https://www.todamateria.com.br/causas-da-revolucao-industrial/#:~:text=As%20causas%20da%20Revolu%C3%A7%C3%A3o%20Industrial,e%20capitais%20dispon%C3%ADveis%20para%20investimento>.

<https://solesw.com.br/o-metodo-cascata-na-gestao-de-projetos/#:~:text=O%20m%C3%A9todo%20cascata%20%C3%A9%20frequentemente,s%C3%A3o%20bem%20definidas%20e%20sequenciais>.

<https://monday.com/blog/pt/desenvolvimento/kanban-x-scrum-qual-e-a-diferenca/#:~:text=No%20Scrum%2C%20todos%20concordam%20com,informa%C3%A7%C3%B5es%20visuais%20do%20quadro%20Kanban>.

<https://scoreplan.com.br/blog/kanban-vs-scrum-entenda-as-diferencas-entre-eles-e-como-usar/#:~:text=J%C3%A1%20o%20Scrum%20prev%C3%AA%20que,produtividade%20alcan%C3%A7ada%20em%20cada%20sprint>.

<https://blog.unopar.com.br/educacao-e-tecnologia/>

<https://blog.brq.com/code-review/#:~:text=A%20revis%C3%A3o%20de%20c%C3%B3digo%20desempenha,entre%20os%20membros%20da%20equipe>.

<https://arphoenix.com.br/documentacao-tecnica-de-software/>