

Universidade São Judas Tadeu Butantã

Gestão e qualidade de software - CCP1AN-BUE1-6507138

Guilherme de Camargo Leite Eubank Pereira - 822142574 - 822142574@ulife.com.br

Guilherme Farias Menoci - 822135941 - 822135941@ulife.com.br

João Henrique Bezerra dos Santos RA: 821141558 - 821141558@ulife.com.br

São Paulo 2024

Tema escolhido: 3

- **Alcançando a Qualidade do Software:**
- **Métodos de Engenharia de Software**

A Engenharia de Software é um campo em constante evolução, com o objetivo de desenvolver softwares de alta qualidade de forma eficiente e eficaz. Os **métodos de engenharia de software** são as técnicas e processos utilizados para alcançar esse objetivo.

O que são Métodos de Engenharia de Software?

Os métodos de engenharia de software são um conjunto de práticas e ferramentas que guiam o desenvolvimento de software desde a concepção até a entrega final. Eles fornecem uma estrutura para organizar o trabalho, garantir a qualidade e controlar os riscos.

Exemplos de métodos:

- **Metodologias Ágeis:** Scrum, Kanban, Extreme Programming (XP) - Focam em entregas incrementais, flexibilidade e colaboração contínua com o cliente.
- **Metodologias Tradicionais:** Waterfall, RUP - Seguem um fluxo sequencial de fases, com planejamento detalhado no início do projeto.
- **Outras Metodologias:** DevOps, Lean Software Development - Combinam práticas de desenvolvimento e operações para acelerar a entrega de software.

- **Técnicas de Gerenciamento de Software**

O gerenciamento de software é um conjunto de práticas e ferramentas que visam planejar, organizar e controlar o desenvolvimento de software, desde a concepção até a entrega final. Existem diversas técnicas e metodologias disponíveis, cada uma com suas próprias características e aplicações.

Por que o gerenciamento de software é importante?

- **Qualidade:** Garante que o software seja desenvolvido com os mais altos padrões de qualidade, atendendo às necessidades do cliente.
- **Eficiência:** Permite otimizar o processo de desenvolvimento, reduzindo custos e tempo de entrega.
- **Gerenciamento de riscos:** Identificar e mitigar os riscos que podem afetar o projeto.
- **Colaboração:** Facilita a comunicação e colaboração entre as equipes envolvidas no projeto.

Principais Técnicas de Gerenciamento de Software

As técnicas de gerenciamento de software podem ser agrupadas em diferentes categorias, como:

Metodologias de Desenvolvimento

- **Waterfall:** Modelo sequencial e linear, onde cada fase é concluída antes de iniciar a próxima. Ideal para projetos com requisitos bem definidos e estáveis.
- **Ágeis:** Focam em entregas incrementais, flexibilidade e colaboração contínua com o cliente. Exemplos: Scrum, Kanban, Extreme Programming (XP).
- **DevOps:** Combina práticas de desenvolvimento e operações para acelerar a entrega de software.

Ferramentas de Gerenciamento

- **Gerenciadores de tarefas:** Trello, Asana, Jira - Organizam tarefas, acompanham o progresso e facilitam a colaboração.
- **Controle de versão:** Git, SVN - Gerenciam as diferentes versões do código fonte, permitindo rastrear mudanças e colaborar em equipe.
- **Integração contínua:** Jenkins, CircleCI - Automatizam a construção, testes e implantação do software.

Técnicas de Engenharia de Software

- **Análise e projeto:** UML, BPMN - Modelam os requisitos e a arquitetura do software.
- **Testes:** Unitários, de integração, de sistema - Verificam a qualidade do software.
- **Documentação:** Requisitos, design, código - Registram as informações sobre o software.

Outras Técnicas

- **Gerenciamento de configuração:** Define e controla as características de um sistema.
- **Gerenciamento de riscos:** Identifica, avalia e mitiga os riscos do projeto.
- **Gerenciamento de qualidade:** Garante que o software atenda aos requisitos de qualidade.

Como escolher a técnica certa?

A escolha da técnica de gerenciamento de software depende de diversos fatores, como:

- **Tamanho e complexidade do projeto**
- **Requisitos do cliente**
- **Equipe de desenvolvimento**
- **Tecnologia utilizada**

Exemplos de Aplicações

- **Desenvolvimento de aplicativos móveis:** Scrum, Kanban, ferramentas de CI/CD.

- **Desenvolvimento de sistemas web:** Waterfall, Agile, ferramentas de versionamento.
- **Manutenção de sistemas legados:** Waterfall, técnicas de refatoração.

Gostaria de se aprofundar em algum tópico específico? Posso fornecer mais detalhes sobre:

- **Metodologias Ágeis:** Scrum, Kanban, XP
- **Ferramentas de gerenciamento:** Jira, Trello, Git
- **Técnicas de teste de software**
- **Gerenciamento de configuração**
- **Casos de uso práticos**

Qual é o seu principal interesse no momento?

controlar o desenvolvimento de software, desde a concepção até a entrega final.

- **Controle de Qualidade**

O que é Controle de Qualidade?

Controle de qualidade (CQ) é um conjunto de atividades e procedimentos que visam garantir que produtos ou serviços atendam a um determinado padrão de qualidade predefinido. Em outras palavras, o CQ assegura que o que é produzido ou oferecido esteja conforme as especificações e atenda às expectativas dos clientes.

Em resumo, o CQ:

- **Verifica:** Se os processos estão funcionando corretamente.
- **Analisa:** Se os produtos ou serviços atendem aos requisitos.
- **Melhora:** Os processos continuam.

Principais Técnicas de Controle de Qualidade

Existem diversas técnicas de CQ, cada uma com suas particularidades e aplicações. Algumas das mais comuns são:

- **Inspeção:** Verificação visual ou por meio de instrumentos de medição para identificar defeitos ou não conformidades.
- **Amostragem:** Análise de uma parte representativa do produto ou serviço para inferir sobre o todo.
- **Controle estatístico de processo (CEP):** Utilização de métodos estatísticos para monitorar e controlar processos.
- **Six Sigma:** Metodologia focada na redução de defeitos e na melhoria de processos.
- **5S:** Ferramenta para organização do ambiente de trabalho.

- **Kaizen:** Melhoria contínua dos processos.

Por que o Controle de Qualidade é Importante?

O CQ é fundamental para o sucesso de qualquer empresa, pois:

- **Aumenta a satisfação do cliente:** Produtos e serviços de qualidade geram clientes mais satisfeitos e fiéis.
- **Reduz custos:** A prevenção de defeitos e a identificação de problemas em estágios iniciais evitam retrabalho e perdas financeiras.
- **Melhora a reputação da empresa:** Uma empresa conhecida por seus produtos e serviços de qualidade tem maior credibilidade no mercado.
- **Aumenta a competitividade:** Empresas com sistemas de CQ eficientes conseguem se destacar da concorrência.

Garantia da Qualidade.

O Software Quality Assurance (SQA) consiste nos meios para garantir a qualidade do software a ser lançado, monitorando os métodos e processos de engenharia de software. O SQA abrange todo o ciclo de vida de desenvolvimento de software incluindo: gerenciamento de requisitos, design de software, codificação, teste.

A SQA desempenha um papel fundamental no desenvolvimento de software, pois ajuda a evitar erros e problemas que podem comprometer a qualidade do produto final. Ao implementar processos de garantia da qualidade, é possível identificar e corrigir falhas e inconsistências antes que elas afetem os usuários finais. A realização de testes de software pode ser feita de forma manual ou automatizada, esta última indica a utilização de ferramentas de teste. As ferramentas de teste permitem ao testador automatizar parte do processo de teste, facilitando a retestagem para validação dos testes.

As principais atividades da SQA são:

- Planejamento da qualidade

É a primeira etapa da SQA, onde são definidos os objetivos e requisitos de qualidade do software. Nessa fase, são estabelecidos os critérios de aceitação e os padrões que serão seguidos durante o desenvolvimento.

- Controle de qualidade

O controle de qualidade envolve a monitorização contínua do processo de desenvolvimento para garantir que os padrões de qualidade sejam cumpridos. Isso inclui a realização de testes e inspeções para identificar possíveis problemas e garantir que o software atenda aos requisitos estabelecidos.

- **Testes de software**

Os testes de software são uma parte essencial, pois permitem verificar se o software funciona corretamente e atende aos requisitos estabelecidos. Isso inclui testes de funcionalidade, desempenho, segurança e usabilidade, entre outros.

- **Revisões de código**

As revisões de código são realizadas para identificar possíveis erros e inconsistências no código fonte do software. Essa atividade é importante para garantir a qualidade do código e evitar problemas futuros.

- **Documentação**

A documentação é uma parte essencial, pois permite que os desenvolvedores e usuários entendam como o software funciona e como utilizá-lo corretamente. Isso inclui a criação de manuais de usuário, documentação técnica e outros materiais relacionados.

Conclusão

O gerenciamento de software estabelece o arcabouço estratégico para o desenvolvimento, definindo metodologias, recursos e prazos. Ao selecionar as técnicas e ferramentas mais adequadas, ele assegura a eficiência e a organização do processo, contribuindo para a entrega bem-sucedida do software.

Em paralelo, o controle de qualidade atua como um guardião da conformidade, verificando se o produto final atende aos requisitos estabelecidos e às normas de qualidade. Ao identificar e corrigir desvios, ele garante que o software seja entregue com a qualidade esperada, minimizando riscos e custos.

A garantia da qualidade de software complementa essa dinâmica, estabelecendo um conjunto de atividades proativas para assegurar que o software seja desenvolvido com qualidade desde o início. Através do planejamento, controle, testes e revisões, a SQA garante a confiabilidade, a usabilidade e a conformidade do software, resultando em uma melhor experiência do usuário e na redução de custos.

Referências :

<https://www.atlassian.com/br/git/tutorials/what-is-version-control#:~:text=Os%20sistemas%20de%20controle%20de,fonte%20ao%20longo%20do%20tempo.>

<https://www.oracle.com/br/devops/what-is-devops/#:~:text=DevOps%20e%20DevSecOps%20fornecem%20um,fornece%20integra%C3%A7%C3%A3o%20e%20entrega%20cont%C3%ADnuas>

<https://www.supero.com.br/blog/metodologias-ageis-mais-usadas/#:~:text=Scrum%2C%20Kanban%2C%20Scrumban%2C%20Extreme,de%20uma%20maneira%2C%20ressaltando%20o>

<https://robsoncamargo.com.br/blog/o-que-e-metodologia-agil#:~:text=A%20metodologia%20%C3%A1gil%20%C3%A9%20uma,em%20entregas%20incrementais%2C%20trabalhando%20em>

<https://fia.com.br/blog/waterfall/#:~:text=A%20metodologia%20Waterfall%20%C3%A9%20um,a%20conclus%C3%A3o%20da%20etapa%20anterior.>

<https://fmxsolucoes.com.br/gestao-de-projetos-de-software/#:~:text=A%20gest%C3%A3o%20de%20projetos%20de%20software%20tamb%C3%A9m%20%C3%A9%20vantajosa%20por,%2C%20parceiros%2C%20fornecedores%20e%20clientes>

<https://www.tecnisys.com.br/garantia-da-qualidade-de-software-sqa/>

<https://ciriusquality.com.br/glossario/o-que-e-sqa-garantia-da-qualidade-de-software/>

<https://www.devmedia.com.br/introducao-a-garantia-de-qualidade-de-software-e-ferramentas-para-teste/28027>

<https://www.globant.com/pt-br/tech-terms/garantia-qualidade-software>

<https://8quali.com.br/o-que-e-controle-de-qualidade-para-que-serve-e-como-fazer-na-minha-empresa/>

<https://www.site.moki.com.br/post/control-de-qualidade>

<https://testingcompany.com.br/blog/conheca-5-beneficios-da-gestao-e-do-control-de-qualidade-de-software>

<https://viniciusgarcia.me/development/entendendo-a-qualidade-do-software/>

<https://www.devmedia.com.br/introducao-ao-gerenciamento-de-qualidade/33435>