

Métricas de Software

Nome do Grupo: InovaTech

Integrantes:

**Gabriel Farah de Lima (RA: 822231424)
(farahzerafacul21@hotmail.com)**

**Webster Diogenes Rodrigues (RA:8222242764)
(rdiogenes.webster12@gmail.com)**

**Bianca Alves Ribeiro (RA: 8222240261)
(bialuno4@gmail.com)**

**Luiz Gustavo França de Abreu (RA: 823210075)
(luizgustavo_40@hotmail.com)**

**Fabício de Barros Narbon (RA: 822227166)
(fabricionarbon50@gmail.com)**

**Rafael Rossetto Guitarrari (RA: 823158602)
(rafaelguitarrari@gmail.com)**

Curso: Gestão e Qualidade de Software

Turma: (CCP1AN-MCD3)

Professor: Robson Calvetti

Resumo dos Artigos

- Roteiro de Métricas do SISP

O conteúdo do documento define regras para padronizar métricas de software no SISP, com foco em melhorar a gestão de projetos, contratos e processos de desenvolvimento e manutenção.

Principais pontos:

- **Objetivo:** Avaliar desempenho, produtividade, qualidade, esforço e custo em projetos e serviços de software.
- **Tipos de Métricas:** Tamanho (ex: Pontos de Função), esforço/custo, qualidade (ex: defeitos), produtividade.
- **Aplicação:** Em todas as fases do ciclo de vida do software, com coleta e análise contínua.
- **Indicadores:** Exemplos incluem custo por ponto de função, índice de retrabalho e cumprimento de prazos.
- **Governança:** Envolve capacitação, definição de responsabilidades e alinhamento com a gestão estratégica de TI.

As métricas visam avaliar contratos, produtividade, qualidade, esforço e custo, apoiando decisões gerenciais. Classificam-se em tamanho, esforço/custo, qualidade e produtividade. Baseiam-se em normas como IFPUG, CMMI e ITIL, e são aplicadas em todas as fases do ciclo de vida do software. Indicadores como custo por ponto de função e índice de retrabalho são usados. A implantação exige responsabilidades definidas, equipes capacitadas e ferramentas de apoio, com governança alinhada à gestão estratégica de TI.

- Guia de Métricas de Software para FINEP

O documento aborda métricas aplicadas a produtos e processos de software, com objetivo em melhorar a qualidade, produtividade e controle dos projetos.

Métricas de Produto

As métricas de produto têm como objetivo avaliar atributos internos do software, como tamanho, complexidade e qualidade do código. Entre as principais abordagens discutidas estão:

- **Tamanho:** Utilização de linhas de código (LOC) e pontos de função (PF) para estimar esforço e custo.

- Complexidade: Destaque para a Complexidade Ciclomática de McCabe, usada para medir a quantidade de caminhos lineares independentes no código-fonte, servindo como base para planejamento de testes.
- Qualidade Interna: Aborda indicadores como número de erros por módulo, densidade de defeitos e cobertura de testes, fundamentais para avaliar a manutenibilidade do sistema.

2. Métricas de Processo

Estas métricas ajudam a entender e melhorar os processos de desenvolvimento, promovendo maior previsibilidade e controle sobre os resultados.

- Produtividade: Medida da relação entre esforço despendido (geralmente em horas/homem) e a quantidade de produto gerado (em LOC ou PF).
- Conformidade ao Cronograma: Avaliação do quanto o projeto está aderente aos prazos planejados.
- Custo e Esforço: Utilização de métricas para comparar o planejado com o realizado, auxiliando na identificação de desvios.

3. Considerações sobre Implantação

O documento também alerta sobre os cuidados necessários para implantar métricas com sucesso. É fundamental envolver a equipe, evitar uso punitivo das métricas e garantir que as informações coletadas sirvam para tomada de decisões e melhorias reais.

- Aplicação de Métricas Ágeis no Desenvolvimento de Software

As metodologias ágeis de desenvolvimento de software são estratégias que valorizam a adaptabilidade, o trabalho em equipe e a entrega contínua de funcionalidades, contrastando com os modelos tradicionais mais rígidos. Elas promovem a interação frequente com o cliente, favorecem a resposta rápida a mudanças e incentivam a melhoria constante dos processos, em vez de seguir um planejamento fixo desde o início do projeto.

Scrum

Organiza o trabalho em ciclos curtos chamados sprints, com funções bem definidas como Product Owner, Scrum Master e o Time de Desenvolvimento, promovendo entregas frequentes e feedback contínuo.

Kanban

Adota um sistema visual (quadro Kanban) para acompanhar o andamento das tarefas, permitindo gerenciar o fluxo de trabalho de forma mais eficiente e identificar gargalos.

Extreme Programming (XP)

Dá ênfase à excelência técnica, práticas de codificação como testes automatizados e integração contínua, e à habilidade de se adaptar rapidamente às mudanças nos requisitos.

Lean Software Development

Baseia-se nos princípios do pensamento Lean, buscando eliminar desperdícios, maximizar o valor entregue ao cliente e melhorar continuamente os processos de desenvolvimento.

- Principais Métricas de Software

As organizações de TI utilizam diversos KPIs (Indicadores-Chave de Desempenho) para monitorar a eficiência do desenvolvimento de software, abrangendo produtividade, desempenho, qualidade e experiência do usuário. Esses indicadores estão agrupados em quatro categorias principais:

1. **Produtividade do Desenvolvedor** – Mede o progresso das equipes por meio de métricas como tempo de entrega, quantidade de código, trabalho em progresso (WIP), velocidade ágil, taxa de sucesso em sprints, número de versões lançadas, entre outras.
2. **Desempenho do Software** – Avalia aspectos como escalabilidade, estabilidade, tempo de resposta, throughput e disponibilidade. Também inclui o conceito de RAS (confiabilidade, disponibilidade e capacidade de assistência).
3. **Defeitos e Segurança** – Engloba métricas como densidade de defeitos, cobertura de código, taxa de detecção de falhas, dívida técnica, moral da equipe, vulnerabilidades e incidentes de segurança, além de tempos médios de detecção, falha e reparo.
4. **Experiência do Usuário (UX)** – Analisa qualitativamente e quantitativamente a interação dos usuários com o sistema, incluindo usabilidade, satisfação, acessibilidade e o Net Promoter Score (NPS).

Essas métricas ajudam as empresas a entender melhor a qualidade do software desenvolvido e tomar decisões estratégicas mais informadas. A escolha dos indicadores deve ser alinhada com os objetivos da organização.