

Universidade São Judas Tadeu - Butantã
Noturno

Nomes: Gabriel Carvalho dos Santos - 821159957
Gustavo Mendes de Lima - 821139542
Matheus dos Santos de Souza - 821133576
Lucas Rodrigues Santos - 823124699

Turma: GQS-CCP1AN-BUE1

Professor: Robson Calvetti

**Ler todos os artigos direcionados pelos hiperlinks
apresentados nos slides deste material;
Resumo dos artigos lidos, no máximo 3 (três) laudas.**

Roteiro de Métricas de Software:

Este documento tem como objetivo apresentar um Guia de Métricas de Software para a Finep, focado no desenvolvimento e manutenção de software, utilizando duas principais métricas: Ponto de Função (PF) e Unidade de Serviço Técnico (UST). A primeira parte do guia detalha um roteiro de contagem de Pontos de Função, baseado no Manual de Práticas de Contagem (CPM 4.3) do IFPUG, e visa complementar esse manual para torná-lo mais aplicável às necessidades específicas da Finep, como em contratos de software. Já a segunda parte introduz a Unidade de Serviço Técnico, uma métrica similar ao Ponto de Função, voltada para demandas específicas de middleware, adaptada a partir de uma métrica criada pela Infraero.

O principal objetivo do guia é apoiar a Finep em sua interação com fornecedores de software, fornecer subsídios para mediações e arbitragens relacionadas à análise dessas métricas, e estabelecer critérios de remuneração para itens mensuráveis e não mensuráveis.

O documento também define que ele poderá ser atualizado sempre que necessário, para se alinhar com novas diretrizes, normas vigentes ou recomendações de órgãos de controle. As atualizações nos documentos de referência, como o Manual de Práticas de Contagem do IFPUG e o Roteiro de Métricas de Software do SISP, também podem levar a modificações neste guia.

Em termos de ordem de precedência, caso haja conflitos ou ambiguidades entre o guia da Finep e outros documentos de referência, as regras estabelecidas neste guia prevalecerão. Quando não houver uma solução específica no guia da Finep, deverá ser seguido o Roteiro SISP, e, por último, o CPM do IFPUG.

O guia visa padronizar a contagem e as estimativas de esforço em projetos de software, garantindo maior consistência e objetividade nos contratos e no acompanhamento dos serviços de desenvolvimento e manutenção de sistemas.

GUIA DE MÉTRICAS DE SOFTWARE FINEP:

Diversas instituições públicas e privadas têm adotado a métrica de Ponto de Função (PF) para estimativas e dimensionamento de projetos de software, devido à sua objetividade, independência da tecnologia utilizada e facilidade de aplicação nas fases iniciais do ciclo de vida do software. A Instrução Normativa SLTI/MP nº 4, de 2014, recomenda a utilização de métricas em contratos de software, restringindo o uso de esforço baseado em homem-hora. Além disso, a Portaria SLTI/MP nº 31, de

2010, sugere que órgãos do SISP adotem o Ponto de Função para a contratação de serviços de desenvolvimento e manutenção de software.

O Tribunal de Contas da União (TCU) também tem incentivado o uso da métrica Ponto de Função Não Ajustado em contratos relacionados ao desenvolvimento de sistemas. Acórdãos específicos recomendam a contagem de Ponto de Função sem ajustes, o uso exclusivo da métrica de tamanho funcional sem vinculação ao esforço em horas, e a distinção entre os custos de desenvolvimento de novas funcionalidades e as alterações em funcionalidades existentes.

A contagem de Pontos de Função é uma técnica para medir o tamanho funcional de sistemas de software, fundamentada no projeto lógico da aplicação. Esse processo se inicia com documentos de requisitos, como o documento de visão, e permite a geração de estimativas de tamanho, prazo e custo do projeto, especialmente nas fases iniciais. No entanto, para lidar com diferentes tipos de manutenção de software, é necessário desenvolver métricas adicionais além das previstas no Manual de Práticas de Contagem (CPM) do IFPUG.

O roteiro de métricas do SISP, baseado nas regras do CPM, foi elaborado para padronizar a contagem de Pontos de Função em contratos de desenvolvimento e manutenção de software, promovendo a objetividade nas medições e estimativas. Este documento também propõe melhorias para garantir a consistência na aplicação das métricas, e destaca a necessidade de um modelo de contratação mais focado em resultados funcionais, ao invés de horas trabalhadas. Além disso, o roteiro sugere mudanças culturais nos órgãos do SISP, orientando sobre como adotar o modelo de contratação baseado no Ponto de Função.

A versão 2.3 do roteiro inclui ajustes nas definições e regras de contagem, especialmente com base nas recomendações da CGU. O objetivo principal do documento é fornecer diretrizes claras e consistentes para a aplicação da métrica em diferentes tipos de projetos de software, facilitando a gestão e o controle de custos e prazos no desenvolvimento de sistemas.

Aplicação de Métricas Ágeis no Desenvolvimento de Software:

A aplicação de métricas ágeis no desenvolvimento de software envolve o uso de indicadores para monitorar e melhorar o desempenho das equipes que adotam metodologias ágeis, como Scrum, Kanban e XP. Essas métricas ajudam a medir o progresso, a produtividade e a qualidade do trabalho realizado, permitindo que as equipes se ajustem continuamente para entregar valor de maneira mais eficiente. Entre as métricas comuns estão a velocidade, que mede a quantidade de trabalho completado durante um sprint, e o lead time, que avalia o tempo total de uma tarefa desde o início até a entrega. O burndown chart acompanha o progresso do sprint, enquanto a taxa de defeitos ajuda a monitorar a qualidade do software ao identificar problemas durante o desenvolvimento. O uso dessas métricas permite que as equipes ajam rapidamente para resolver problemas, melhorem o processo de desenvolvimento e garantam a entrega constante de valor ao cliente. O objetivo é

fornecer dados objetivos para decisões mais informadas e para um ciclo de feedback contínuo.

23 métricas de desenvolvimento de software que devem ser monitoradas:

O artigo discute 23 métricas essenciais para monitorar o desenvolvimento de software, divididas em quatro categorias principais: produtividade do desenvolvedor, desempenho do software, defeitos e segurança, e experiência do usuário (UX). Essas métricas ajudam a avaliar o progresso do desenvolvimento, a qualidade do software e a satisfação dos usuários, além de fornecerem insights valiosos para melhorar a eficiência e a performance do processo de criação e manutenção dos aplicativos.

Na produtividade do desenvolvedor, são abordadas métricas como o tempo de entrega, que mede o tempo total necessário para concluir um projeto; a quantidade de código (KLOC), que mede o volume de código produzido; o trabalho em progresso (WIP), que indica o quanto a equipe já concluiu de um projeto; e a velocidade ágil, que avalia o número de pontos de histórias completados por sprint. Além disso, a taxa de sucesso da meta Sprint e o número de versões de software também são fundamentais para monitorar o progresso e os lançamentos do software.

O desempenho do software é avaliado com base em métricas que analisam a eficiência do sistema, como escalabilidade, tempo de resposta e disponibilidade. A métrica de desempenho (throughput) mede a quantidade de dados processados por unidade de tempo, enquanto a RAS (reliability, availability, serviceability) indica a confiabilidade, disponibilidade e facilidade de manutenção do sistema.

As métricas de defeitos e segurança incluem a densidade de defeitos, que calcula o número de falhas por mil linhas de código, e a cobertura de código, que mede o percentual de código coberto por testes automatizados. A taxa de detecção de defeitos indica a eficácia na identificação de problemas antes do lançamento, enquanto a dívida técnica avalia o custo acumulado devido à falta de resolução de problemas durante o desenvolvimento. Além disso, a vulnerabilidade de segurança e os incidentes de segurança reais monitoram as falhas de segurança no sistema, com ênfase na rapidez de detecção e resolução de problemas.

Por fim, as métricas de usabilidade e experiência do usuário (UX) medem a satisfação do usuário com o software, analisando aspectos como a facilidade de uso e a eficiência do software para atingir objetivos. A métrica de Net Promoter Score (NPS), que avalia a probabilidade de um usuário recomendar o software a outros, é um indicador chave da experiência do usuário.

Essas métricas fornecem uma visão abrangente da qualidade do software e do processo de desenvolvimento, ajudando as equipes a melhorar continuamente tanto o produto quanto sua produtividade e desempenho.

