



UNIVERSIDADE FEDERAL DO ESTADO DO RIO DE JANEIRO
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
ESCOLA DE INFORMÁTICA APLICADA

Definição de um Conjunto de Indicadores para Apoiar a Gestão de Projetos

Cesar Luis Barbosa da Silva
William Luiz Neves Brum

Orientador
Gleison dos Santos Souza

RIO DE JANEIRO, RJ – BRASIL
JUNHO DE 2016

Definição de um Conjunto de Indicadores para Apoiar a Gestão de Projetos

Cesar Luis Barbosa da Silva

William Luiz Neves Brum

Projeto de Graduação apresentado à Escola de
Informática Aplicada da Universidade Federal do
Estado do Rio de Janeiro (UNIRIO) para obtenção do
título de Bacharel em Sistemas de Informação.

Aprovada por:

Prof. Gleison dos Santos Souza, D.Sc. (UNIRIO)

Leonardo Azevedo, D.Sc. (UNIRIO)

Cristina Teles Cerdeiral, D.Sc. (UNIRIO)

RIO DE JANEIRO, RJ – BRASIL.

Agradecimentos

À minha família, em especial minha mãe Leila e minha tia Neide, e Heloá pelo apoio incondicional e motivação desde sempre.

Ao nosso orientador Gleison Santos, que sempre nos motivou a melhorar e buscar mais conhecimento, além da dedicação e paciência necessárias e essenciais para a evolução deste trabalho.

Aos professores e técnicos administrativos da UNIRIO, por sempre fornecerem o apoio e base necessários para que eu chegasse aqui.

Agradeço os professores Leonardo Azevedo e Cristina Cerdeiral, por avaliarem esse trabalho.

Ao meu companheiro neste trabalho Cesar Barbosa, por todas as horas de estudo e ajuda mútua que passamos nos últimos anos.

Ao João Felipe e Sylvio Bonelli por arranjarem um tempo em suas agendas para as entrevistas necessárias para esse trabalho.

Aos meus colegas de faculdade, destacando Carlos Magno, Victor Azevedo, Lucas Souza, Guilherme Caeiro, Bernardo Gouvea, Antônio França, Karina Martinez e Caroline Loppi por dividir comigo esses períodos da faculdade com muito estresse, felicidade e troca de conhecimento.

Por fim, agradeço às pessoas com quem convivi no trabalho, pois a cada dia eu aprendo algo novo e estão sempre me motivando a me tornar um profissional cada vez melhor. Em particular, gostaria de agradecer a João Manoel, Marcos Mele, Fillipe Silva e Carlos Vetter.

William Luiz Neves Brum

Agradeço a **minha família** por me dar todo o apoio para poder chegar onde estou e me tornar o que me tornei.

Agradeço a minha avó **Ruth** e meu avô **Manoel**, por todo o suporte, e colaboração na minha educação e formação de vida e pelo amor e carinho dados a mim desde sempre.

A meu pai, **Carlos Roberto**, muito obrigado por tudo que sempre fez, se eu me tornar um pai e homem da forma que o meu é para mim, em todos os momentos, certamente já serei o melhor do mundo. Obrigado por ser meu exemplo e por todo o carinho, atenção e dedicação.

A minha mãe, **Célia**, maior referência e meu amor, muito obrigado, por toda a força e luta sempre e por me mostrar o mundo da melhor forma possível. Obrigado por toda a confiança, estímulo e carinho. Você é a melhor mãe do mundo!

A minha irmã, **Flávia**, obrigado pelo amor, confiança e companheirismo e, aos meus irmãos, **Igor** e **Gustavo**, obrigado pelos momentos ainda juntos, pelos conselhos e pelo apoio. Saibam que podem contar comigo para sempre.

A minha companheira **Ana Caroline**, pelo carinho, amor, lealdade e dedicação. Obrigado por ser minha companheira da vida e meu amor. Obrigado pela paciência nos momentos de estudo e trabalho e pelo carinho e consideração sempre.

Agradeço ao nosso orientador, **professor Gleison Santos**, pela orientação exemplar e por nos guiar na busca da escrita de um texto de excelência. Mesmo em meio a um semestre de grande tormenta de minha parte conseguiu nos dar o suporte necessário a confecção deste trabalho de conclusão. Obrigado por tudo.

Agradeço a banca responsável pela avaliação deste trabalho, o professor **Leonardo Azevedo**, professor do curso de Sistemas de Informação da UNIRIO, e a professora **Cristina Cerdeiral** e no ensejo agradeço a **todos professores e colaboradores** do BSI.

Agradeço aos prezados **João Felipe** e **Sylvio Bonelli** pelo tempo despendido para participação nas entrevistas e pela consideração e apoio.

Agradeço ao meu amigo e colega de faculdade, o qual divido este trabalho, **William Brum**, pela amizade e pela consideração. Pelas horas e madrugadas no desenvolvimento em conjunto deste trabalho e pelo foco e determinação em realizarmos um excelente trabalho.

Agradeço aos amigos que o BSI me trouxe, **Antônio França**, **Bernardo Gouvêa**, **Caroline Loppi**, **Guilherme Caeiro**, **Karina Martinez**, **Lucas Souza**, pelo companheirismo, amizade e torço que esta amizade perdure mesmo em breve sendo egressos.

Cesar Luis Barbosa da Silva

RESUMO

A plataforma Github é utilizada por diversos desenvolvedores não só para atrair possíveis colaboradores para os mais diversos projetos, mas também é utilizada por gerentes de projeto e suas equipes como plataforma de hospedagem de código. O monitoramento dos projetos, apoiado pela medição de software, é uma parte essencial para atingir o sucesso. Todavia, a criação de planos de medição pode ser uma tarefa complexa e, no contexto do Github, os indicadores existentes não suprem todas as necessidades dos gerentes de projeto.

A proposta deste trabalho é definir um conjunto de indicadores que complementem aqueles já disponibilizados pelo Github e padronizá-los através da aplicação do *template* AMP.br. Foram realizadas duas avaliações a fim de avaliar a qualidade dos indicadores descritos seguindo o AMP.br que serviram de base para possíveis melhorias a ser aplicadas ao modelo, bem como a elaboração de novos indicadores.

Palavras-chave: Medição de Software, Indicadores, Gerenciamento de Projetos, Github, Open Source, MR-MPS-SW

ABSTRACT

Github platform is used by many developers not only to attract potential partners for various projects, but it is also used by project managers and their teams as a code hosting platform. Project monitoring supported by software measurement is an essential part to achieve success in software projects. However, the definition of measurement plans can be complex and, in the context of Github, existing indicators do not attend all project managers' needs.

Our goal is to define a set of indicators that can complement those of Github and standardize them by applying the AMP.br template. We performed two evaluations in order to assess the quality of the indicators. We were able to identify improvements that can be applied to the model as well as new indicators.

Keywords: Software Measurement, Indicators, Project Management, Github, Open

Source, MR-MPS-SW

Índice

1. Introdução	1
1.1 Contexto	1
1.2 Motivação.....	2
1.3 Objetivos	4
1.4 Organização do texto.....	4
2. Revisão Bibliográfica.....	6
2.1 Medição de Software.....	6
2.2 Indicadores	7
2.3 MR-MPS-SW	8
2.4 Github.....	10
2.5 Template AMP.br.....	13
2.5.1 Componente Perfil Organizacional.....	13
2.5.2 Componente Mapeamento de Indicadores	14
2.5.3 Componente Fichas.....	17
2.6 Considerações Finais.....	27
3. Indicadores para Uso com o Github	28
3.1 Aplicação para coleta de dados do Github.....	28
3.2 Formulação dos Indicadores Complementares	31
3.3 Aplicação do Template AMP.br	34
3.3.1 Construção do Mapa de Objetivos	35
3.3.2 Construção do Mapa de Indicadores	36
3.3.3 Construção do Mapa do MR-MPS-SW	37
3.3.4 Construção da Ficha de Indicador.....	40
3.4 Considerações Finais.....	50
4. Avaliação dos Indicadores Construídos.....	51
4.1 Introdução	51

4.2 Método de Pesquisa para o Estudo de Caso.....	52
4.2.1 Elaboração do questionário	52
4.2.2 Identificação de um projeto ativo de desenvolvimento de software no Github 54	
4.2.3 Análise do perfil profissional dos participantes	55
4.2.4 Procedimento de Entrevista com o Desenvolvedor	55
4.2.5 Análise dos resultados da Avaliação com o Desenvolvedor.....	56
4.2.6 Análise da entrevista com o Gerente de projeto	58
4.3 Considerações Finais.....	59
5. Conclusão.....	60
5.1 Considerações Finais.....	60
5.2 Limitações do Trabalho.....	60
5.3 Trabalhos Futuros	61
6. Referências Bibliográficas	62
7. Apêndice 1 – Aplicação PyNdicators	66
8. Apêndice 2 – Conjunto de Indicadores Criados	70

Índice de Tabelas

Tabela 1 - Notação para o grau de relacionamento	19
Tabela 2 - Critérios para uso das categorias selecionadas.....	20
Tabela 3 - Relacionamento entre os Objetivos Estratégicos, de Medição e as Necessidades de Informação	36
Table 4 - Relacionamento entre necessidades de informação e indicadores.....	36
Table 5 - Perfis dos Participantes	55
Tabela 6 - Avaliação de Aplicabilidade preenchida pelo entrevistado	57

Índice de Figuras

Figure 1 - Indicador do Github de linhas inseridas e removidas por colaborador	11
Figura 2 - Identificador da Organização (BONELLI , 2014)	14
Figura 3 - Mapa de Objetivos	15
Figura 4 - Mapa do MR-MPS-SW	15
Figura 5 - Representação de um indicador no mapa de Indicadores	16
Figura 6 - Elementos da notação para indicação do relacionamento entre indicadores (BONELLI , 2014)	16
Figura 7 - Mapa de Indicadores	17
Figura 8 - <i>Template</i> para Registro de Indicadores (BONELLI , 2014).....	18
Figura 9 - Fragmento da Ficha de Indicador referente ao nome do indicador	18
Figura 10 - Fragmento da Ficha de Indicador referente a necessidade de informação ..	19
Figura 11 - Fragmento da Ficha de Indicador referente a categorias	19
Figura 12 - Fragmento da Ficha de indicador referente aos Objetivos de Medição e seus tipos	21
Figura 13 - Fragmento da Ficha de indicador referente ao nível MR-MPS-SW.....	21
Figura 14 - Exemplo de representação gráfica de um indicador	22
Figura 15 - Fragmento da Ficha de Indicador referente a área de procedimento de medição e análise.....	22
Figura 16 - Notação utilizada para descrever o Procedimento de Medição e Análise (BONELLI , 2014)	23
Figura 17 - Exemplo da seção procedimento de medição e análise preenchida.....	24
Figura 18 - Fragmento da Ficha de Indicador referente ao procedimento de análise baseado em critérios (BONELLI , 2014).....	25
Figura 19 - Exemplo da seção Procedimento de Análise Baseado em Relacionamentos preenchida.....	25
Figura 20 - Exemplo de uma Ficha de Indicador preenchida.....	26
Figura 21 - Aplicação de Extração de Estatísticas sobre linhas de código.....	29
Figura 22 - Exemplo do produto final da extração de estatísticas sobre linhas de código	31
Figura 23 - Mapa de objetivos	38
Figura 24 - Mapa de Indicadores	39
Figura 25 - Ficha do Indicador RIC	41

Figura 26 - Ficha do Indicador IAF.....	42
Figura 27 - Categorias do Indicador RIC	43
Figura 28 - Categorias do Indicador IAF.....	43
Figura 29 - Nível MR-MPS-SW dos indicadores RIC e IAF.....	44
Figura 30 - Gráfico do Indicador RIC	45
Figura 31 - Gráfico do Indicador IAF	45
Figura 32 - Procedimento de Medição e Análise do indicador RIC.....	46
Figura 33 - Procedimento de Medição e Análise do indicador IAF.....	46
Figura 34 - Procedimento de Medição - RIC	47
Figura 35 - Procedimento de Análise - RIC	47
Figura 36 - Procedimento de Análise Baseado em Critérios - RIC.....	48
Figura 37 - Procedimento de Análise Baseado em Critérios - IAF.....	48
Figura 38 - Procedimento de Análise Baseado em Relacionamentos - RIC	49
Figura 39 - Procedimento de Análise Baseado em Relacionamentos - IAF	49
Figure 40 - Termo de Consentimento para participação do estudo de caso	53
Figure 41 - Avaliação de aplicabilidade do estudo de caso.....	54

1. Introdução

1.1 Contexto

Segundo BASS *et al.* (1999), a medição de software é uma avaliação quantitativa de software de quaisquer aspectos de processos e produtos de software que permite o seu melhor entendimento e, a partir disso, permite um melhor planejamento, controle e melhoria daquilo que é produzido e da sua forma de produção.

Para compreender melhor o que é a medição, primeiramente é necessário entender o que são medidas. Conforme ROCHA *et al.* (2012), medidas caracterizam uma propriedade de um objeto em termos quantitativos, além de fornecer informações quantitativas que auxiliam na tomada de decisão de negócio. Ainda de acordo com ROCHA *et al.* (2012), o objetivo principal da medição só é atingido quando as informações coletadas através da medição são utilizadas para direcionar as ações necessárias às organizações e seus projetos. Além disso, a medição é crucial para os gerentes de projeto, pois auxilia no monitoramento do andamento do desenvolvimento de software, tanto para resolver problemas de desvio de planejamento, quanto para demonstrar resultados para os seus superiores e colegas de trabalho (BONELLI, 2014).

Além disso, existem os indicadores que, segundo BARCELLOS (2009), são medidas base ou derivadas que, ao serem associadas a critérios de avaliação ou decisão pré-definidos, são capazes de fornecer informações que auxiliam no alcance de um objetivo estabelecido. Um exemplo de um indicador seria um gráfico que apresenta o número de defeitos de código detectados em um determinado período de tempo. Os indicadores representam produtos de informação que atendem as necessidades de informação inicialmente identificadas e são registrados nos planos de medição, que incluem (ROCHA *et al.*, 2012):

- Objetivos da Medição;
- Necessidades da Informação;
- Medidas e Indicadores;
- Definição operacional para as medidas e indicadores, incluindo:
 - i. Nome da Medida

- ii. Definição da Medida
- iii. Mnemônico
- iv. Tipo de Medida
- v. Entidade de Medida
- vi. Propriedade de Medida
- vii. Unidade de Medida
- viii. Tipo de Escala
- ix. Valores da Escala
- x. Intervalo esperado dos dados
- xi. Fórmula de cálculo da medida
- xii. Procedimento de Medição
- xiii. Momento da Medição
- xiv. Periodicidade da Medição
- xv. Responsável da Medição
- xvi. Procedimento de Análise da Medição
- xvii. Momento da Análise da Medição
- xviii. Periodicidade da Análise da Medição
- xix. Responsável pela Análise da Medição

1.2 Motivação

Empresas necessitam ter uma visão gerencial bem definida de seus processos de desenvolvimento para a tomada de decisão. Para BASILI (2014), o uso de planos de medição é essencial para a garantia de que os processos estão sendo seguidos corretamente. Além disso, também são úteis para que possa ser observado, por meio de indicadores, quais pontos do processo não estão de acordo com o previsto, além do que pode ser melhorado.

Segundo BASILI (2014), a realidade do alinhamento organizacional em diversas organizações é severa. Situações como imposição de diferentes objetivos sem prioridades bem definidas, reorganizações criando ineficiência, objetivos e estratégias de longo prazo não sendo sistematicamente seguidas, rastreadas, avaliadas e atualizadas são comuns em diversas organizações. Conforme BASILI (2014), um alinhamento organizacional bem sucedido por ser alcançado por meio da aplicação de um conjunto

básico de princípios para formar uma conexão entre objetivos e estratégias em toda a organização e integrá-los no contexto de negócio.

Gerentes de projeto de desenvolvimento de software também encontram diversos problemas ao criar um plano de medição sem utilizar um padrão bem definido, tornando o processo mais trabalhoso e menos padronizado. Devido à falta de padronização de uma lista de indicadores alinhados com os objetivos da organização, é necessária uma busca pelos indicadores mais adequados para atender às necessidades de informação de cada projeto. Consequentemente, não há uma padronização dos planos de medição e resulta em gerentes de diferentes projetos abordarem problemas similares de maneiras diferentes. (BONELLI, 2014).

Ademais, projetos de desenvolvimento de software utilizam repositórios online com a finalidade de não só manter cópias e evoluções do código fonte desenvolvido, mas também com o intuito de colaboração entre os desenvolvedores. Um dos serviços de hospedagem de repositórios mais utilizados é o Github, que se destaca pela sua funcionalidade de *forking*, isto é, copiar um repositório de um usuário para outro. Isso permite que colaboradores copiem o repositório de um usuário, façam modificações e depois as enviem para o usuário original. É natural, então, que equipes de desenvolvimento utilizem essa plataforma para seus projetos e os gerentes, consequentemente, se interessem nas informações fornecidas pelo Github para acompanhamento dos colaboradores.

O *template* desenvolvido por BONELLI (2014), chamado de “Apoio à Medição de Projetos de Software – Uma abordagem Baseada em Relacionamentos” (AMP.br), tem como objetivo propor o registro dos indicadores por meio de fichas que organizam a descrição e mapas de relacionamentos do indicador, facilitando a seleção dos indicadores para os projetos de uma organização, levando em conta os objetivos que desejam ser alcançados (BONELLI, 2014).

Dessa forma, a motivação do trabalho está ligada aos benefícios que podem ser proporcionados por um plano de medição bem construído, com a definição e formalização de indicadores relevantes, dentro do contexto de uso da plataforma do Github para gerência de projetos de software.

1.3 Objetivos

Este trabalho tem como objetivo a definição de indicadores complementares aos oferecidos pelo Github adequados para gerência de projetos nesta plataforma e, além disso, padronizá-los utilizando o *template* AMP.br.

A partir dessa abordagem, pretende-se identificar também:

- i. Objetivos de Medição de Software adequados ao uso em projetos *open source* hospedados no Github;
- ii. Procedimentos de análise para alcançar os objetivos de software;
- iii. Objetivos de medição relativos aos cenários considerados;
- iv. Questões de medição e indicadores a partir dos objetivos de medição.

Dessa maneira, busca-se mostrar como essa abordagem auxilia na construção de um plano de medição visando ao alinhamento com os objetivos de uma organização.

1.4 Organização do texto

O presente trabalho está estruturado em capítulos e, além desta introdução, será desenvolvido da seguinte forma:

- Capítulo II: Contém a Revisão Bibliográfica do trabalho, apresentando os principais conceitos abordados no trabalho. Inicialmente são abordados conceitos relacionados a medição de software e indicadores, Github e a linguagem de programação Python. Também é apresentada a proposta de BONELLI (2014) para o apoio a gerenciamento de projetos de software.
- Capítulo III: Aborda como a coleta de dados do Github foi realizada, bem como os indicadores identificados e formalizados com o uso do *template* AMP.br.
- Capítulo IV: Apresenta os resultados de um estudo de caso feito em cima dos indicadores produzidos no contexto deste trabalho. Foram entrevistados um membro de equipe de desenvolvimento de software que utiliza o Github como plataforma de desenvolvimento e um gerente de projetos com experiência na elaboração e análise de indicadores relacionados a projetos de software.

- Capítulo V: Conclusões – Reúne as considerações finais, assinala as contribuições da pesquisa e sugere possibilidades de aprofundamento posterior.

2. Revisão Bibliográfica

Nesse capítulo são abordados os principais conceitos envolvidos no trabalho: Medição de Software e Indicadores; MR-MPS-SW e, por fim, o Github e o template AMP.br.

2.1 Medição de Software

Na fase de execução do gerenciamento do projeto, é importante o monitoramento e controle de ações relevantes para que o projeto atinja os objetivos a que se propõe (Santos, 2009). Buscando o aprimoramento das práticas de Engenharia de Software e, consequentemente o desenvolvimento de produtos de melhor qualidade em projetos, as organizações têm mostrado um crescente interesse por programas de melhoria de processos. (WEBER e LAYMAN, 2002; CANFORA *et al.*, 2004; DUMKE *et al.*, 2006 *apud* BARCELLOS, 2009). Para tal, é necessário encontrar meios que permitam controlar e monitorar melhor o projeto. A gestão de projetos somente atinge determinado nível de eficácia e exatidão se houver métricas que possibilitem gerenciar por meio de fatos (FERNANDES, 1995 *apud* SANTOS, 2009).

Em Engenharia de Software, isso remete à necessidade de que cada atividade seja planejada e a sua execução seja objetivamente controlada. Com isso, as medições realizam um papel importante no desenvolvimento de software efetivo e eficiente (FENTON e PFLEEGER, 1997, *apud* MONTEIRO, 2008; KAN, 2003 *apud* MONTEIRO, 2008).

A medição, seja em qualquer tipo de projeto, contribui para melhor entendimento dos processos que fazem parte de um projeto, ajuda a estabelecer pontos importantes no projeto, contribui na análise do *status* do projeto de acordo com o que foi planejado e permite avaliar até que ponto o processo que está sendo executado encontra-se de acordo com os padrões especificados (SANTOS, 2009).

Segundo BARCELLOS (2009), a medição de software é uma avaliação quantitativa de qualquer aspecto dos processos e produtos da Engenharia de Software, que permite seu melhor entendimento e, com isso, auxilia o planejamento, controle e melhoria do que se produz e de como é produzido (BASS *et al.*, 1999 *apud* BARCELLOS, 2009). O elemento básico da medição, que propicia a análise quantitativa, são as medidas. Elas caracterizam, em termos quantitativos, alguma propriedade de um objeto da Engenharia de Software (BASILI e ROMBACH, 1994 *apud* BARCELLOS, 2009).

2.2 Indicadores

Segundo ISO/IEC (2002), MCGARRY *et al.* (2002), SEI (2004) (*apud* Monteiro, 2008), um indicador é uma medida ou uma combinação de medidas, normalmente apresentado na forma de gráfico ou tabela, que provê entendimento a respeito de uma questão ou conceito de software. Para BARCELLOS (2009), uma medida que esteja sendo utilizada para indicar o alcance de um objetivo desempenha o papel de indicador e é relevante à tomada de decisão.

Os indicadores podem ainda ser descritos como a base para a análise e tomada de decisões (MONTEIRO e OLIVEIRA, 2011 *apud* BONELLI, 2014). Ou seja, tomadores de decisão devem ter acesso aos indicadores para obterem informações que embasem suas decisões. TAKARA, BETTIN e TOLEDO (2007) (*apud* MONTEIRO, 2008) complementam essa visão, ressaltando que um indicador é medida extremamente importante para a organização, sendo acompanhada e analisada periodicamente a partir de uma meta de desempenho pré-estabelecida.

Sendo assim, indicadores são medidas que, ao serem associadas a critérios de avaliação ou decisão pré-definidos, são capazes de fornecer informações que auxiliam no alcance de um objetivo estabelecido (BARCELLOS, 2009). Os indicadores escolhidos são a base para a análise e tomada de decisão, portanto são eles que devem ser apresentados aos usuários de medição.

O PMBOK (PMI, 2013) orienta a utilização de indicadores nas atividades de monitoramento de projetos, para que as organizações possam acompanhar o desempenho de prazos e custos, assim como os riscos e a própria equipe. Os indicadores utilizados pelo gerente de projetos para monitorar seu projeto são registrados no Plano

de Medição do Projeto. O produto final do planejamento da medição em um projeto ou organização é um Plano de Medição (BARCELLOS, 2009)

2.3 MR-MPS-SW

O programa MPS.BR é um programa coordenado pela Associação para Promoção da Excelência do Software Brasileiro (SOFTEX) com o apoio do MCTI (Ministério da Ciência, Tecnologia e Inovação) o FINEP, o SEBRAE e o Banco Interamericano de Desenvolvimento.

O objetivo do programa MPS.BR é aumentar a competitividade das organizações pela melhoria dos seus processos. De acordo com o Guia Geral MPS de Software (2016), a sigla MPS é uma marca genérica associada aos Modelos MPS, compreendendo as siglas MPS-SW associada à Melhoria de Processo de Software, a sigla MPS-SV associada à Melhoria de Processo de Serviços e a sigla MPS-RH associada à melhoria de Processo de Recursos Humanos.

Para um setor de software competitivo nacional e internacionalmente, é importante que a eficiência e eficácia dos processos sejam o foco das empresas desse nicho, e isto basicamente ocorre a partir de correlações com modelos e padrões internacionais de qualidade, o que aumenta, além da competitividade, a produtividade nestas empresas. Para tal, o modelo MPS para software (MR-MPS-SW) se mostra adequado ao perfil de empresas com diferentes tamanhos e características e espera-se que o modelo MPS para software seja compatível com os padrões de qualidade aceitos internacionalmente e que tenha como pressuposto o aproveitamento de toda a competência existente nos padrões e modelos de melhoria de processo já disponíveis.

Particularmente nesse trabalho, o foco de análise é o componente Modelo de Referência MPS para Software (MR-MPS-SW).

De acordo com o Guia Geral MPS de Software (2016), O MR-MPS-SW (Modelo MPS para software) baseia-se nos conceitos de maturidade e capacidade de processo para a avaliação e melhoria da qualidade e produtividade de software e serviços correlatos e também para a melhoria da qualidade e produtividade dos serviços prestados. Dentro desse contexto, ele possui quatro componentes: Modelo de Referência MPS para Software (MR-MPS-SW), Modelo de Referência MPS para Serviços (MR-

MPS-SV), Método de Avaliação (MA-MPS) e Modelo de Negócio para Melhoria de Processo de Software e Serviços.

O Modelo de Referência MPS para Software (MR-MPS-SW) define níveis de maturidade que são uma combinação entre processos e sua capacidade. A definição dos processos segue os requisitos para um modelo de referência de processo apresentados na ISO/IEC 15504-2 (2002), declarando o propósito e os resultados esperados de sua execução. Isso permite avaliar e atribuir graus de efetividade na execução dos processos. A capacidade do processo é a caracterização da habilidade do processo para alcançar os objetivos de negócio, atuais e futuros; estando relacionada com o atendimento aos atributos de processo associados aos processos de cada nível de maturidade.

Os níveis são acumulativos, ou seja, se a organização está no nível F, esta possui o nível de capacidade do nível F que inclui os atributos de processo dos níveis G e F para todos os processos relacionados no nível de maturidade F (que também inclui os processos de nível G). Isto significa que, ao passar do nível G para o nível F, os processos do nível de maturidade G passam a ser executados no nível de capacidade correspondente ao nível F. Em outras palavras, na passagem para um nível de maturidade superior, os processos anteriormente implementados devem passar a ser executados no nível de capacidade exigido neste nível superior.

De forma a ilustrar e relacionar, neste trabalho, observam-se dois níveis de maturidade do MR-MPS-SW, o nível G e o nível F.

O nível G de maturidade é o nível “parcialmente gerenciado”, é composto pelos processos Gerência de Projetos e Gerência de Requisitos. O processo observado neste trabalho é o Gerência de Projetos (GPR). O processo GPR tem como propósito estabelecer e manter planos que definem as atividades, recursos e responsabilidades do projeto, bem como prover informações sobre o andamento do projeto que permitam a realização de correções quando houver desvios significativos no desempenho do projeto. O propósito deste processo evolui à medida que a organização cresce em maturidade.

O outro nível de maturidade observado no trabalho é o nível F. O nível F de maturidade é o nível “gerenciado” composto pelos processos do nível de maturidade anterior (G) acrescidos dos processos Aquisição, Garantia da Qualidade, Gerência de Configuração, Gerência de Portfólio de Projetos e Medição. O processo observado neste trabalho é o Gerência de Configuração (GCO). O processo GCO tem como propósito

estabelecer e manter a integridade de todos os produtos de trabalho de um processo ou projeto e disponibilizá-los a todos os envolvidos.

2.4 Github

O Github¹ é uma plataforma de hospedagem de código voltada para controle de versão e colaboração, na qual diversos desenvolvedores podem trabalhar em projetos simultaneamente de forma colaborativa e, por meio de repositórios, compartilharem alterações de códigos ou envio de arquivos para o repositório. As principais funcionalidades do Github, além de hospedar o código, são *forking*, *pull requests* e *social networking*.

O *forking* é a ação de criar um projeto baseado em um já existente, isto é, permitir que desenvolvedores trabalhem em um mesmo projeto em repositórios diferentes de tal maneira que um desenvolvedor pode realizar alterações e utilizar o projeto reformulado como um novo projeto. Além disso, o projeto originado pelo *fork* pode ainda ser atualizado conforme o projeto original recebe alterações.

Os *pull requests* são originados a partir de colaboradores que realizaram *fork* de um projeto, realizaram diversas alterações e querem que os desenvolvedores originais avaliem as mudanças realizadas e as incluam no projeto original. Através do *pull request*, o desenvolvedor do *fork* e o desenvolvedor original podem se comunicar com a finalidade de realizar alterações colaborativas.

Outra funcionalidade importante é o *social networking* do Github. Cada usuário possui um perfil, que é bastante similar a um currículo, que contém todas as contribuições daquele usuário através de *pull requests* ou repositórios que criou ou participou.

A análise dos dados utilizados no desenvolvimento desse trabalho foi realizada por meio da extração de informações sobre *issues* e colaboradores de repositórios hospedados no Github.

Uma funcionalidade importante do Github é o registro de *issues*, que podem ser entendidas como tarefas ou problemas no projeto, dependendo de como o projeto é gerido. As *issues* podem ser classificadas em diversos rótulos, como, por exemplo, melhoria ou defeito, com a finalidade de não só destacar qual o tipo daquela issue, mas

¹ <http://www.github.com>

também auxiliar o gerente do projeto no rastreamento dos tipos de atividades desempenhadas pelos colaboradores.

Outra seção importante do Github é a visualização dos *commits*. A ação de enviar algo para o repositório, seja uma alteração de uma única linha de código ou envio de diversos arquivos de documentação, é chamada de commit.

A Figura 1 apresenta um exemplo de indicador provido pelo Github (linhas de código inseridas e removidas por colaborador).

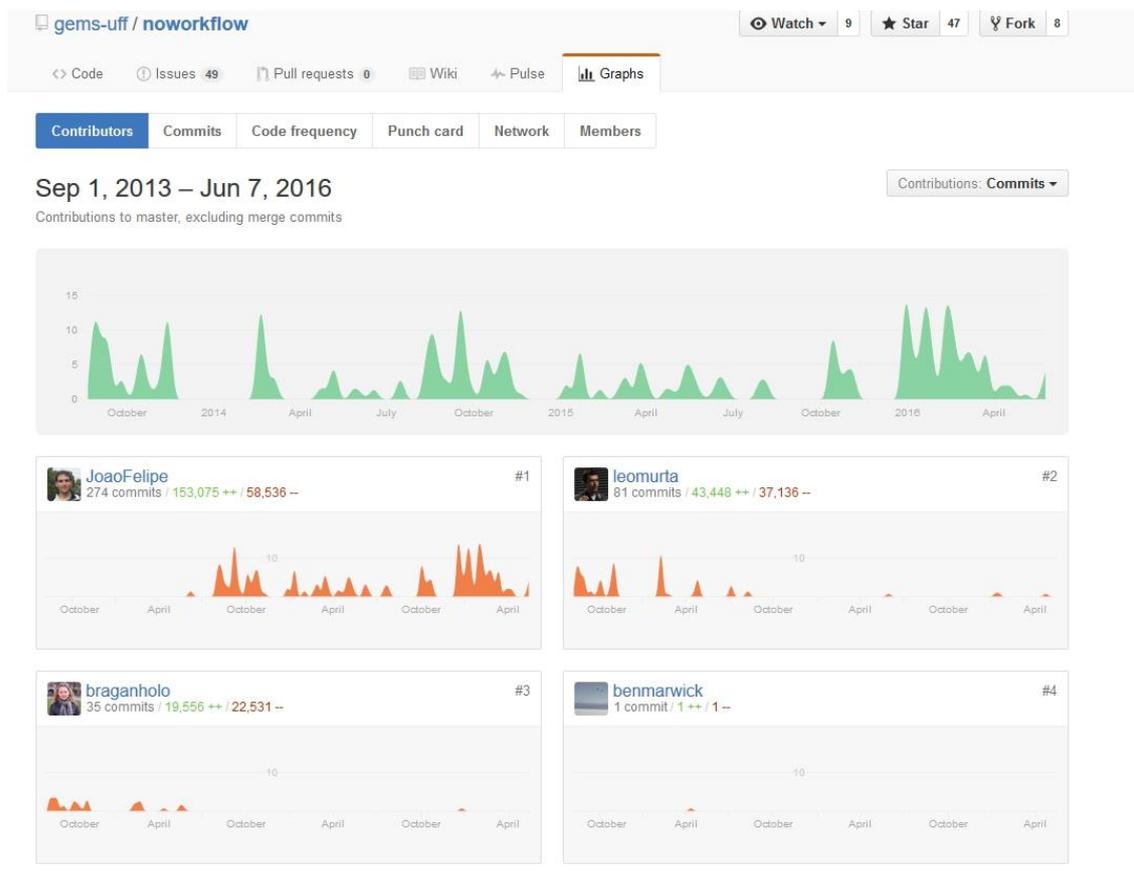


Figure 1 - Indicador do Github de linhas inseridas e removidas por colaborador

Os outros indicadores fornecidos pelo Github são:

- Número de *commits* por semana;
- Frequência de código - épocas do ano com maior número de inclusões e remoções de linhas de código;
- *Punch Card* – Horas do Dia e Dias da Semana com a maior frequência de *commits*;
- *Network* – *Commits* realizados nas diferentes árvores do projeto.

Os indicadores do Github, apesar de apresentarem informações interessantes relacionadas à produtividade e desempenho dos colaboradores, não atendem a certos

aspectos do Github como as *issues*. Portanto, neste trabalho, procurou-se desenvolver indicadores que complementem e, em alguns casos, substituam os indicadores disponíveis pela plataforma.

O Github possui uma *Application Programming Interface* (API) que permite desenvolvedores acessar diversos dados de repositórios, tais como: chamados, estatísticas sobre entregas de código, atividade de colaboradores e informações de versões da aplicação. O acesso é feito através de *Hyper Text Transfer Protocol Secure* (HTTPS) e é obtido através do endereço: <https://api.github.com>. As principais seções da API utilizadas nesse trabalho foram as seções de *Issues*, *Commits* e *Statistics*.

A seção *Issues* (<https://developer.github.com/v3/issues/>) fornece dados referentes aos diferentes tipos de *issues* de um repositório, como o nome da issue, seu estado (por exemplo, aberta ou fechada), a mensagem que descreve a issue, rótulo, o responsável e as datas de abertura e fechamento da issue.

A seção *Commits* (<https://developer.github.com/v3/repos/commits/>), por sua vez, fornece os dados dos *commits* realizados pelos colaboradores de um repositório. Exemplos de informações disponíveis nessa seção são: autor do commit, mensagem do commit e data do commit.

Por fim, a seção *Statistics* (<https://developer.github.com/v3/repos/statistics/>) fornece dados que permitem diferentes visualizações de atividade em um repositório. São exemplos de dados dessa seção: nome do colaborador, número de linhas adicionadas em uma semana, número de linhas removidas em uma semana e a semana do ano.

Todos os dados são enviados e recebidos através de Notação de Objetos JavaScript (JSON). O JSON tem como base um subconjunto da linguagem de programação JavaScript e é independente de linguagem para uso, pois utiliza diversas convenções comuns às linguagens como C, C++, C#, Java e Python. Logo, é ideal para a troca de dados entre aplicações.

Com o propósito de manipular os dados recebidos em JSON, foi escolhida para este trabalho uma biblioteca em Python que conseguisse tratar os dados corretamente e exportá-los para um formato de fácil acesso aos gerentes de projeto, como CSV ou XLS. O pacote escolhido foi o Github3², que é centralizado em ter uma forte organização lógica dos métodos necessários para a interação com a API do GitHub. O

² <https://github3py.readthedocs.io/en/master/>

Github3 permite usuários utilizarem todas as funções da API do GitHub com facilidade através da linguagem Python. Além disso, foi utilizado o pacote CSV que realiza uma análise sintática dos resultados obtidos pelo Github3 e constrói um arquivo no formato CSV baseado neles, seguindo um padrão de colunas e linhas para representar cada registro obtido.

2.5 Template AMP.br

Com o objetivo de estruturar as informações coletadas dos indicadores foi utilizado o *template* desenvolvido por BONELLI (2014), que é dividido em três partes: Perfil Organizacional, Mapeamento de Indicadores e Fichas de Indicadores.

2.5.1 Componente Perfil Organizacional

Segundo BONELLI (2014), o componente *Perfil Organizacional* é utilizado para indicar as características da organização que utilizará o AMP.br na representação de seus indicadores e é composto pelas seguintes seções: Nome da Organização, Repositório de Medição e Versão do Documento (vide Figura 2). A finalidade do repositório é centralizar os documentos e dados que serão utilizados na composição dos indicadores, sejam os documentos planilhas ou dados obtidos a partir de bancos de dados. A estruturação do repositório de dados é diferente entre organizações, portanto o componente precisa armazenar qual o tipo de repositório é utilizado por uma determinada organização.

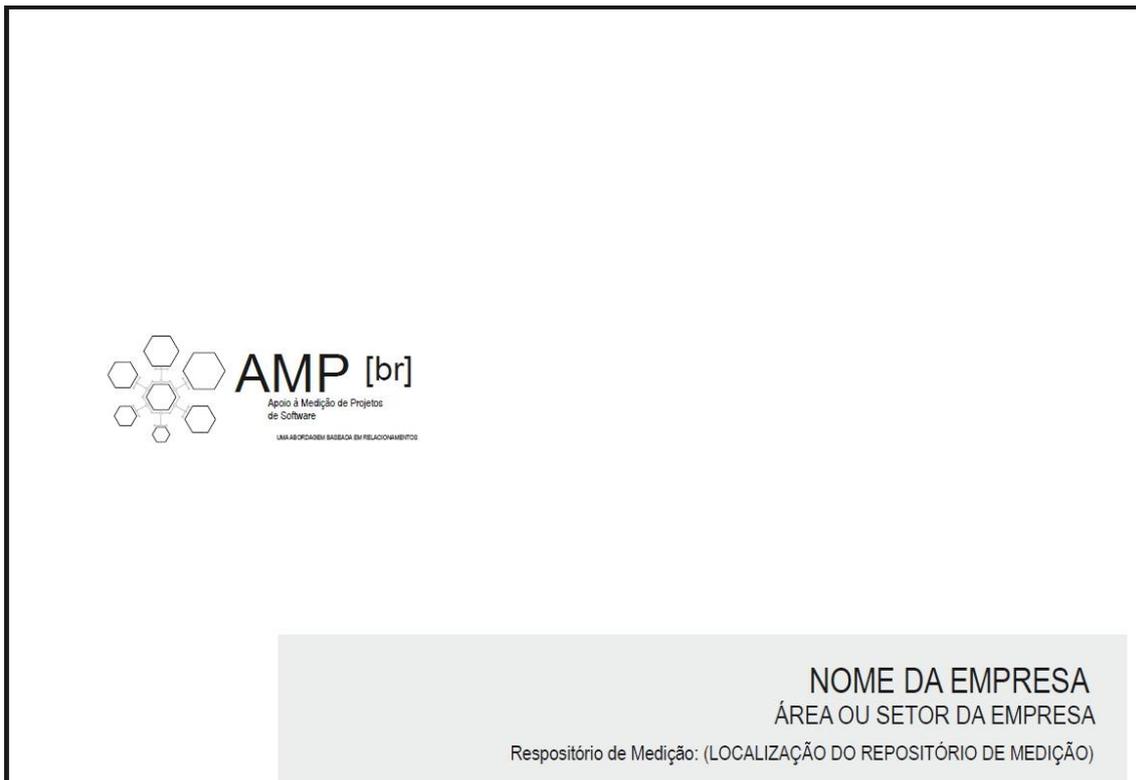


Figura 2 - Identificador da Organização (BONELLI , 2014)

2.5.2 Componente Mapeamento de Indicadores

O componente seguinte, *Mapeamento dos Indicadores*, propõe a criação de três mapas: Mapa de Objetivos e Indicadores, Mapa para o MR-MPS-SW e Mapa de Indicadores. Os mapas, conforme BONELLI (2014), tem como objetivo auxiliar o gerente de projetos a selecionar os indicadores de acordo com os objetivos que devem ser monitorados, os níveis do MR-MPS-SW que devem ser contemplados e os indicadores e medidas necessários para realizar as análises.

O *Mapa de Objetivos e Indicadores* é responsável por fornecer uma visibilidade ao gerente de projeto sobre quais indicadores são importantes a fim de alcançar os objetivos da organização. O mapa é composto por objetivos estratégicos da organização, que por sua vez são a base da formação dos objetivos de medição. A partir desses últimos são identificadas as necessidades de informação e, por fim, os indicadores são formados a partir dessas necessidades. Logo, segundo BONELLI (2014), quando um indicador está associado a uma necessidade de informação, que por sua vez está associada a um objetivo de medição, pode-se concluir que o indicador é utilizado para monitorar um objetivo de medição. Na Figura 3 pode ser observado um exemplo do Mapa de Objetivos e Indicadores preenchidos. Os dados apresentados nesse *template*,

assim como os apresentados nos próximos exemplos, foram construídos durante o Estudo de Caso a ser apresentado no Capítulo III.

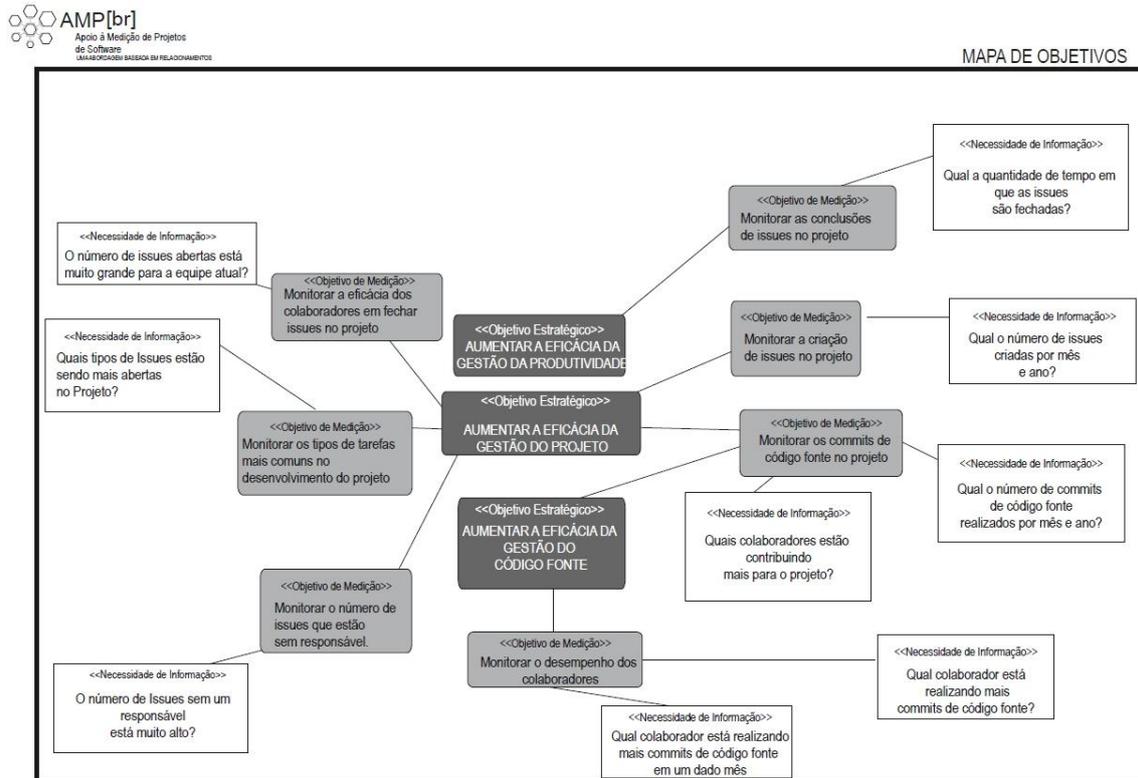


Figura 3 - Mapa de Objetivos

A Figura 4 apresenta um exemplo do Mapa para o MR-MPS-SW, no qual as duas primeiras colunas representam a sigla e o nome dos indicadores, enquanto nas colunas seguintes estão os níveis do MR-MPS-SW entre G e F e o relacionamento, caso exista, dos indicadores com os processos através da marcação de um X com o fundo cinza. O objetivo do mapa é, através da marcação, identificar quais os indicadores são mais adequados para apoiar cada processo de cada nível e, conseqüentemente, auxiliar na seleção dos indicadores necessários de acordo com o nível de maturidade da organização (BONELLI, 2014).

INDICADORES DE PATTERN		MR-MPS-SW	
		G	F
SIGLA	NOME	GPR	GCO
RIC	RÓTULO DE ISSUES MAIS COMUNS	X	
IAF	ISSUES ABERTAS X FECHADAS	X	
NIA	NÚMERO DE ISSUES COM X SEM RESPONSÁVEL	X	
TFI	TEMPO PARA FECHAMENTO DE ISSUES	X	
NIC	NÚMERO DE ISSUES CRIADOS POR MÊS / ANO	X	
CMA	COMMITTS POR MÊS / ANO		X
IEA	INCLUSÕES X EXCLUSÕES POR AUTOR		X
PCA	PORCENTAGEM DE COMMITTS POR AUTOR		X
RCM	RANKING DE COLABORADOR POR MÊS		X

Figura 4 - Mapa do MR-MPS-SW

O *Mapa de Indicadores*, conforme BONELLI (2014), tem como objetivo apresentar o relacionamento entre os indicadores considerando algumas categorias preenchidas. Foram consideradas para esse trabalho as seguintes categorias: Esforço, Escopo, Tempo, Custo, Qualidade e Produtividade. Cada indicador é representado em um retângulo, conforme mostra a Figura 5. Também é representado o relacionamento entre o indicador com uma de suas categorias, que será explorado mais a frente.



Figura 5 - Representação de um indicador no mapa de Indicadores

BONELLI utilizou uma notação baseada em mapas conceituais de Kane e Trochim (2007) para representar os relacionamentos no Mapa de Indicadores, pois “é uma representação comum na documentação da relação de causa e efeito” (BONELLI, 2014). Além disso, foram adicionados os seguintes elementos nessa notação (Figura 6):

Elemento	Descrição
	Indica uma variação que é entendida como positiva no indicador.
	Indica uma variação que é entendida como negativa no indicador.
	Indica um atalho no mapa de relacionamentos, onde N é o número do atalho. É utilizado para amenizar uma eventual poluição visual do mapa, ao substituir longas linhas e setas por atalhos.

Figura 6 - Elementos da notação para indicação do relacionamento entre indicadores (BONELLI, 2014)

Na Figura 7 é apresentado um exemplo do Mapa de Indicadores relacionando os indicadores desenvolvidos.

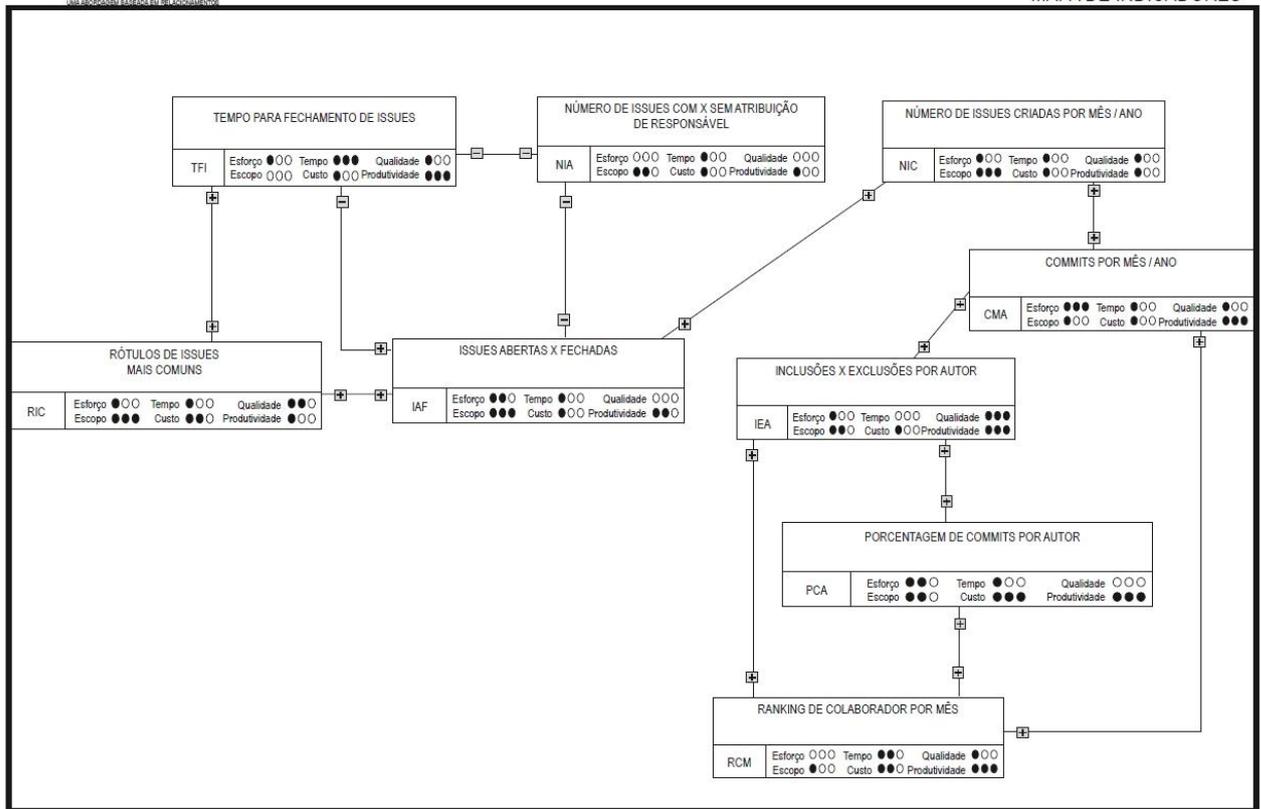


Figura 7 - Mapa de Indicadores

2.5.3 Componente Fichas

A finalidade das fichas de indicadores é apresentar todas as informações importantes sobre um indicador em uma única página (ficha), com o objetivo de simplificar o acesso às informações de cada indicador, segundo BONELLI (2014). As fichas são chamadas de *Fichas de Registro de Indicador* e são divididas em diversas áreas, cada uma contendo informações específicas daquele indicador. A Figura 8 apresenta um *template* da ficha e, sem seguida, cada elemento presente nela é explicado.

Figura 8 - Template para Registro de Indicadores (BONELLI , 2014)

A. *Indicador*

Nessa área são apresentadas as seguintes informações: nome do Indicador, sigla e descrição, conforme a Figura 9. Segundo BONELLI (2014), as informações devem ser concisas e diretas, com a finalidade de facilitar o entendimento e seleção do indicador pelo gerente de projetos. A sigla deve ser formada por uma combinação única de caracteres dentro do conjunto de indicadores.

Indicador
RIC – Rótulos de Issues Mais Comuns
Acompanha o andamento do projeto sob o aspecto de tipos de Issues. Permite ao gerente visualizar qual tipo de tarefa está sendo trabalhada com maior frequência

Figura 9 - Fragmento da Ficha de Indicador referente ao nome do indicador

B. *Necessidade de informação*

Nessa seção se encontra um questionamento do gerente de projetos que é respondido através do indicador, auxiliando na tomada de decisão. A Figura 10 consiste de um exemplo dessa seção

Necessidade de Informação

Quais tipos de Issues estão sendo mais abertas no Projeto?

Figura 10 - Fragmento da Ficha de Indicador referente a necessidade de informação

C. Categoria

A seção de categoria apresenta, conforme mencionado previamente, a relação entre o indicador e as categorias escolhidas para avaliação, bem como o grau de relacionamento entre eles. As categorias podem ser visualizadas na Figura 11:

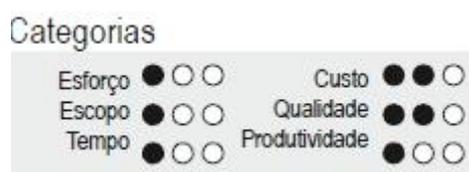


Figura 11 - Fragmento da Ficha de Indicador referente a categorias

As categorias selecionadas por BONELLI (2014) foram baseadas nas propostas do PSM (MCGARRY *et al.*, 2002). Contudo, após adaptações em algumas categorias do PSM, as categorias tornaram-se as mesmas propostas por MONTEIRO (2008). O grau de relacionamento pode ser representado de acordo com a Tabela 1

Tabela 1 - Notação para o grau de relacionamento

Grau de relacionamento com a categoria		Descrição do relacionamento	Exemplo
● ● ●	Impacto Direto	O indicador está claramente relacionado com a categoria. Algumas vezes o próprio nome do indicador pode dar indícios dessa relação.	<i>Indicador:</i> IAF – ISSUES ABERTAS X ISSUES FECHADAS <i>Categoria:</i> PRODUTIVIDADE
● ● ○	Influência Direta	Quando o indicador possui uma relação de causa e efeito clara com indicadores de outra categoria. Esse relacionamento pode ser percebido, por exemplo, pelo uso de medidas compartilhadas entre indicadores.	<i>Indicador:</i> PCA – PORCENTAGEM DE COMMITTS POR AUTOR <i>Categoria:</i> ESFORÇO

Grau de relacionamento com a categoria		Descrição do relacionamento	Exemplo
● ○ ○	Influência Indireta	Quando não há uma relação de causa e efeito claramente definida, mas sob determinadas condições é possível perceber algum tipo de impacto entre o resultado do indicador e os indicadores da categoria.	<i>Indicador:</i> NIA - NÚMERO DE <i>ISSUES</i> COM X SEM RESPONSÁVEL <i>Categoria:</i> TEMPO
○ ○ ○	Sem Influência identificada	Não há uma relação conceitual, de causa e efeito ou de compartilhamento de medidas identificadas.	<i>Indicador:</i> TFI -TEMPO PARA FECHAMENTO DE <i>ISSUES</i> <i>Categoria:</i> ESCOPO

Conforme BONELLI (2014), vale ressaltar que um indicador pode estar presente em mais de uma categoria, isto é, possuir diversos graus de relacionamento com todas as categorias. A Tabela 2 apresenta os critérios de classificação de cada categoria selecionada:

Tabela 2 - Critérios para uso das categorias selecionadas

Categoria	Critérios
Produtividade	O indicador está relacionado com a produtividade do projeto. Auxiliam o indicador a representar se os valores da produtividade estão de acordo com o esperado
Tempo	Indicadores que se baseiam em prazo, relacionados a distribuição em períodos de tempo ou atrasos
Custo	Indicadores que se baseiam em valores monetários, representando em como um determinado indicador afeta o custo do projeto
Escopo	Indicadores que são responsáveis pelo acompanhamento de requisitos do projeto, indicando volatilidade, distribuição por tipos de requisitos ou variações de tamanho
Qualidade	Indicadores relacionados aos aspectos da qualidade de um projeto, como cobertura de testes, densidade de defeitos ou tempo de falha
Esforço	Monitoramento do esforço necessária para a execução das tarefas do projeto

D. *Objetivos*

São considerados dois tipos de objetivos: objetivos estratégicos e objetivos de medição. O objetivo estratégico é a intenção com a qual as ações estratégicas, segundo BONELLI (2014), são planejadas e realizadas. Por exemplo, “aumentar a eficácia da gestão do projeto”. Por outro lado, um objetivo de medição representa como ações relacionadas a medição de software são planejadas e realizadas, como “Monitorar os *commits* de código fonte no projeto”.

Ademais, os objetivos de medição (Figura 12) ainda podem ser classificados em três tipos, segundo BARCELLOS *et al* (2013): objetivo de monitoramento e controle de projetos (por exemplo: “Monitorar as criações de *issues* no projeto”), objetivo de medição de qualidade (um exemplo seria “reduzir o número de defeitos no projeto”) ou objetivo de medição de desempenho (por exemplo: “monitorar os *commits* de código fonte no projeto”). As informações sobre os tipos de objetivos também devem, segundo BONELLI (2014), auxiliar na tomada de decisão sobre quais indicadores devem fazer parte do plano de medição.

Objetivo de Medição	Tipo
Monitorar os tipos de tarefas mais comuns no desenvolvimento do projeto	<input checked="" type="checkbox"/> controle <input type="checkbox"/> qualidade <input type="checkbox"/> desempenho

Figura 12 - Fragmento da Ficha de indicador referente aos Objetivos de Medição e seus tipos

E. Nível MR-MPS-SW

Essa seção do *template* apresenta ao gerente de projetos o nível MR-MPS-SW do indicador para apoio à tomada de decisão baseada nos resultados esperados do indicador (Figura 13).

MR-MPS-SW	
NÍVEIS	G
PROC.	GPR

Figura 13 - Fragmento da Ficha de indicador referente ao nível MR-MPS-SW

Na Figura 13, *PROC.* é a abreviação para processo, identificando os processos que possuem resultados esperados e que podem ser apoiados com o uso do indicador descrito (BONELLI, 2014). Além disso, é sugerido por BONELLI (2014) que sejam usadas as siglas utilizadas para identificar os processos do MR-MPS-SW, como, por exemplo, GPR – Gerência de Projetos e GCO – Gerência de Configuração.

F. Representação Gráfica

Um indicador é frequentemente apresentado de forma gráfica através da plotagem dos dados coletados visando uma maneira fácil e conveniente de visualização dos dados. Portanto, segundo BONELLI (2014), os *template* possui uma área destinada

a gráficos na Ficha do Indicador. A representação gráfica é utilizada para auxiliar o gerente de projeto a identificar, de uma maneira fácil e intuitiva, se um indicador atende as suas necessidades. A Figura 14 contém um exemplo de representação gráfica de um indicador.

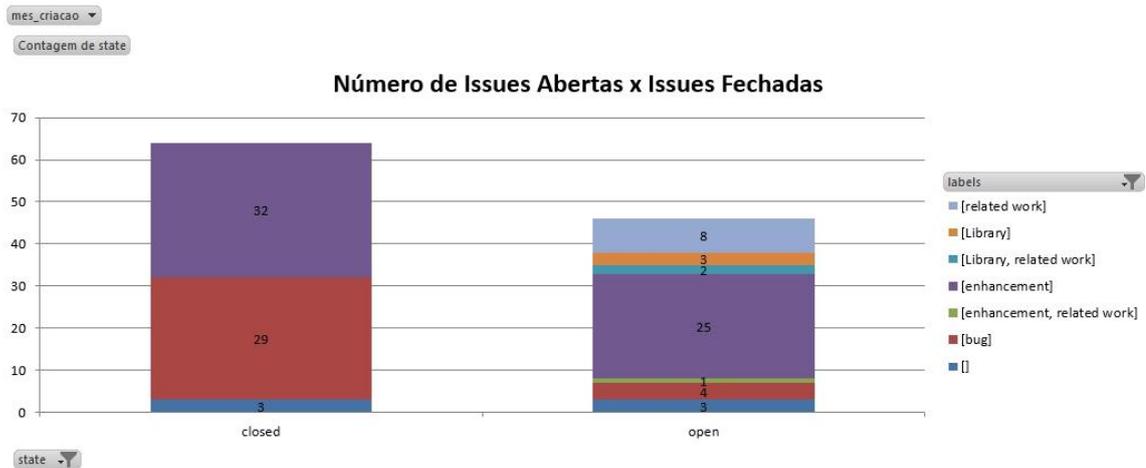


Figura 14 - Exemplo de representação gráfica de um indicador

G. Procedimento de Medição e Análise

Conforme BONELLI (2014), essa seção do template é responsável por apresentar o fluxo de coleta de dados e análise do indicador, isto é, como e onde os dados devem ser adquiridos e qual a maneira apropriada de análise desses dados. O procedimento é, portanto, dividido em duas etapas: Medição e Análise (Figura 15).

Procedimento de Medição e Análise			
MEDICÃO			
ANÁLISE			
Medida Base: COBERTURA DE TESTES	Periodicidade: UMA VEZ POR ITERAÇÃO	Responsáveis: Coleta: @responsável/Tester	Análise: GP

Figura 15 - Fragmento da Ficha de Indicador referente a área de procedimento de medição e análise

O fluxo construído nessa seção utiliza os seguintes elementos do BPMN (BPMN, 2014), apresentados na Figura 16:

Elemento Visual	Nome	Definição
[Texto]	Elemento Textual	São exibidos próximos a outro elemento para definir alguma informação complementar. <i>Início do Fluxo</i> , <i>Decisão</i> e <i>Seta</i> utilizam esse elemento para prover informações adicionais ao entendimento do gerente do projeto.
 [Elemento Textual]	Início do Fluxo	Todo procedimento deve iniciar com este elemento e abaixo dele deve haver um <i>Elemento Textual</i> com o nome da ação que motivou o início do fluxo, ou seja, aqui é apresentado o momento da medição.
	Final do Fluxo	Pode haver mais de um final possível. É importante que fique claramente descrito no fluxo qual é seu término ou termos possíveis.
	Atividade Simples	Dentro deste elemento é representada a atividade que deve ser realizada em um determinado momento do procedimento.
	Atividade Composta	Quando necessário que a descrição de uma atividade inclua outras atividades, este elemento deve ser utilizado para agrupar todas as atividades.
	Atividade Automática	Atividades que sejam executadas de forma automática, por exemplo, por planilhas ou sistemas, devem ser representadas por este elemento.
	Decisão	Indica que neste passo do procedimento deve ser tomada uma decisão. A decisão é descrita como um comentário associado a esse elemento. As opções são representadas por múltiplas setas indicando a sequência de atividades para cada opção. A seta deve ter um <i>Elemento Textual</i> que indique qual a opção da decisão permite seguir sua direção.
	Comentário	São informações complementares para esclarecer o procedimento (opcional). Utilizado pelo elemento <i>Decisão</i> .
	Seta	Indica o fluxo a ser seguido após um elemento.
	Artefato	Um artefato é um elemento visual que indica o que uma atividade deve receber como entrada para ser executada. Também pode explicitar uma saída específica de uma atividade.

Figura 16 - Notação utilizada para descrever o Procedimento de Medição e Análise (BONELLI, 2014)

A Figura 17 apresenta um exemplo sobre como essa seção é devidamente preenchida. Além disso, há informações complementares sugeridas por BONELLI (2014), como *Medida Base*, *Periodicidade* e *Responsáveis de Coleta e Análise*. A *Medida Base* é responsável por indicar quais medidas são necessárias para compor o indicador descrito; *Periodicidade* é qual a frequência de tempo na qual a medição deve ser realizada (por exemplo, mensalmente ou semanalmente); por fim, os *Responsáveis de Coleta e Análise* têm a função de definir quais os papéis do projeto possuem a tarefa de, respectivamente, coletar os indicadores e analisá-los.

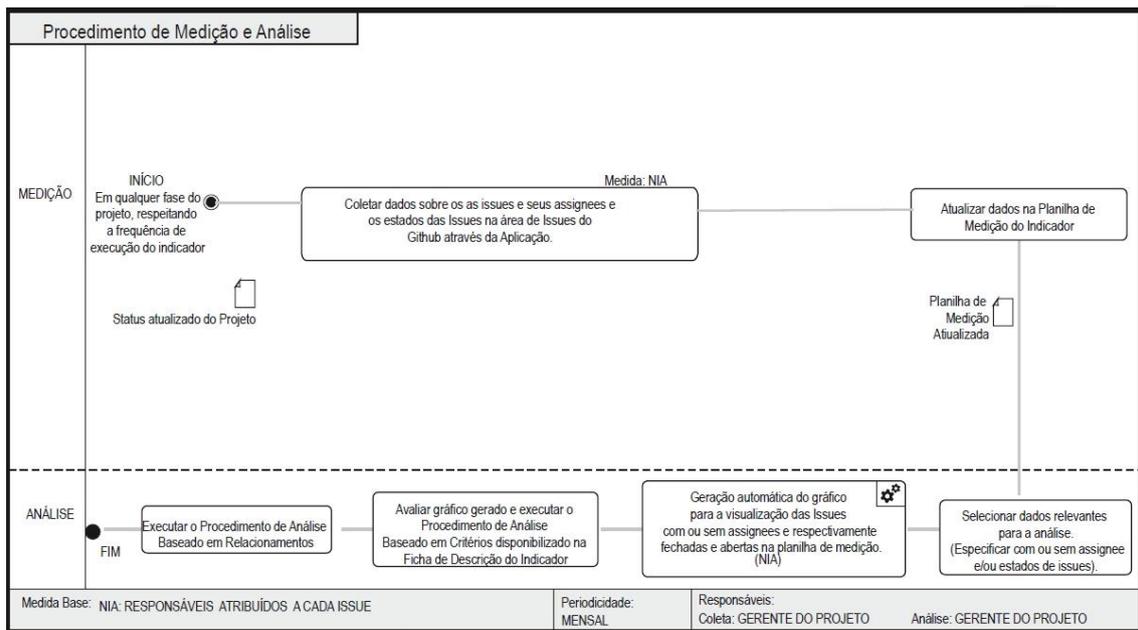


Figura 17 - Exemplo da seção procedimento de medição e análise preenchida

H. Procedimento de Análise Baseado em Critérios.

Para BONELLI (2014), o objetivo dessa seção é definir quais critérios devem ser avaliados pelo gestor do projeto ao realizar a análise de um determinado indicador utilizando a ficha de indicadores. Os critérios são baseados em: *Premissa*, *Possível Conclusão* e *Possíveis Ações* (vide Figura 18). A *Premissa* descreve uma possível análise do gráfico, por exemplo uma coluna do gráfico estar se sobressaindo em relação as outras. A *Possível Conclusão* indica uma possível interpretação do gerente de projetos sobre aquela premissa; por fim, as *Possíveis Ações* representam possíveis soluções caso a premissa seja validada.

Procedimento de Análise Baseado em Critérios			
ID	Premisa	Possível Conclusão	Possíveis Ações

Figura 18 - Fragmento da Ficha de Indicador referente ao procedimento de análise baseado em critérios (BONELLI , 2014)

I. *Procedimento de Análise Baseado em Relacionamentos*

O último elemento do *template* consiste no relacionamento entre os indicadores (Figura 19), ou seja, como um indicador influencia e é influenciado por outros. Os relacionamentos dessa seção são os mesmos apresentados no Mapa de Indicadores.

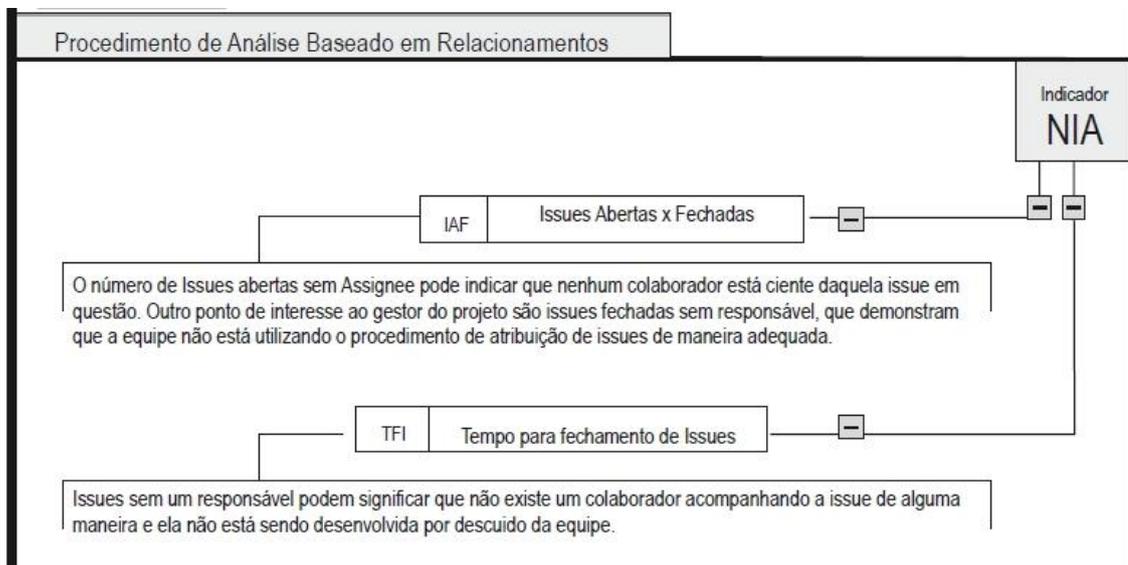


Figura 19 - Exemplo da seção Procedimento de Análise Baseado em Relacionamentos preenchida

Um exemplo de uma Ficha de Registro de Indicadores preenchida é apresentado na Figura 20 com o indicador *Issues* Abertas x *Issues* Fechadas.

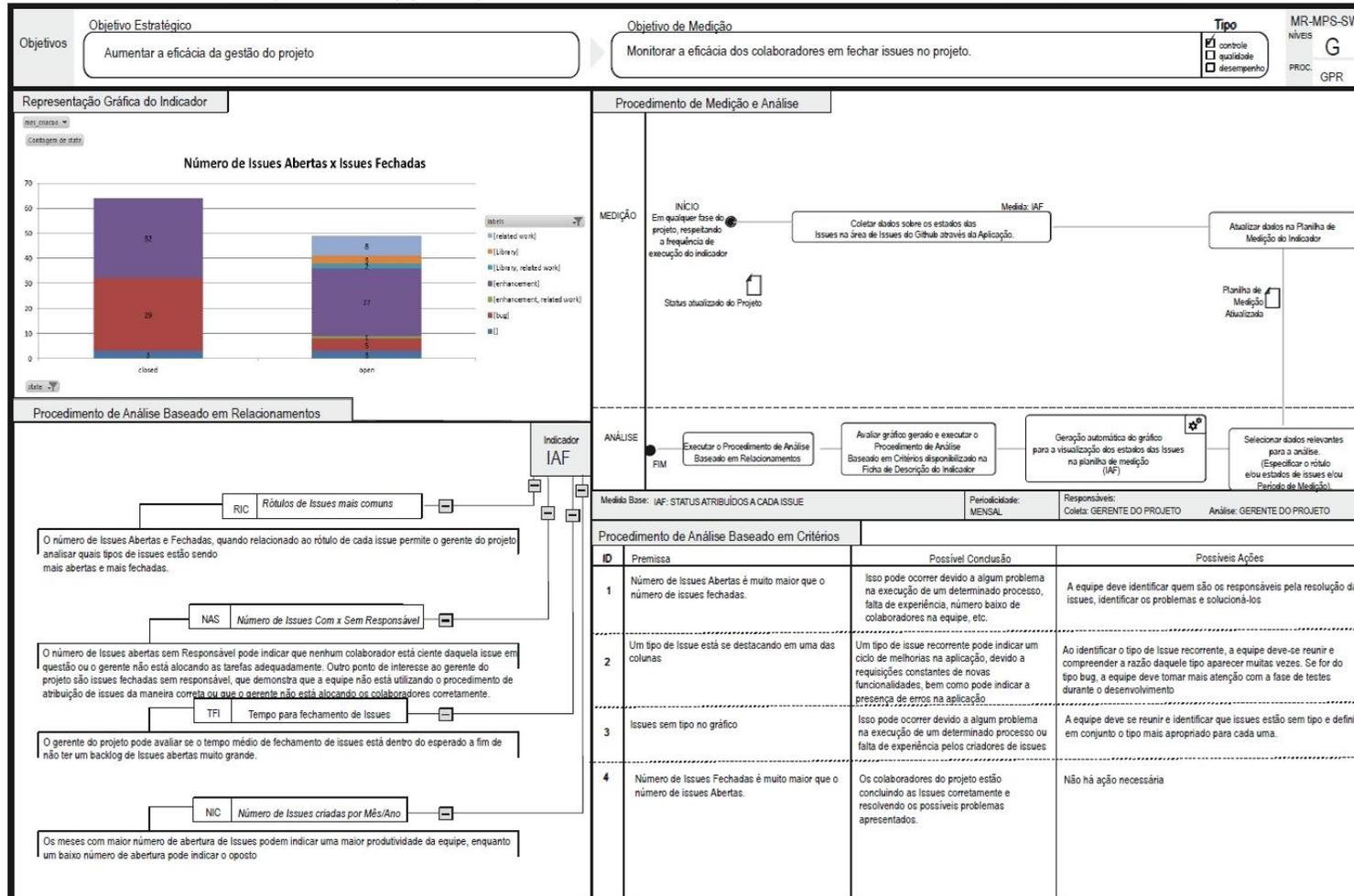


Figura 20 - Exemplo de uma Ficha de Indicador preenchida

2.6 Considerações Finais

Nesse capítulo foram abordados: uma visão geral sobre Medição de Software e Indicadores, seguidos pelo Github e o pacote *github3* utilizado para desenvolver a aplicação. Por fim, foi abordado o *Template* AMP.br (BONELLI, 2014) e os componentes pelos quais é formado, além de como tais componentes devem ser preenchidos. No próximo capítulo serão abordados os indicadores construídos para complementar os já existentes pelo Github e como o AMP.br foi aplicado para a formalização desses indicadores.

3. Indicadores para Uso com o Github

Este capítulo apresenta como a coleta de dados do Github é realizada por meio da aplicação desenvolvida para este trabalho e como o template AMP.br foi preenchido com essas informações.

3.1 Aplicação para coleta de dados do Github

O objetivo deste trabalho é elaborar indicadores para o Github a fim de complementar os já existentes e padronizá-los por meio da aplicação do *template AMP.br*. A expectativa é que os indicadores propostos possam substituir ou serem usados em conjunto com os já existentes e apresentar para os gerentes de projeto e equipe de desenvolvimento uma maneira simplificada de compreensão dos indicadores através do AMP.br.

O estudo de caso descrito nesse trabalho consiste na análise dos dados de projetos hospedados na plataforma Github e construção de fichas e mapas de indicadores a fim de elaborar um Plano de Medição para as equipes e gerentes que utilizam o Github, visando a melhorias em seus processos de desenvolvimento de software. Para coletar os dados foi desenvolvida uma aplicação em Python, chamada de *PyNdicators*, utilizando o pacote *github3* para acessar a API do Github e obter os dados necessários para a elaboração dos indicadores.

As seções da API do Github que foram utilizadas nesse trabalho foram as seções *Issues*, *Commits* e *Statistics*. As seções foram escolhidas por abrangerem os principais dados para a composição de indicadores para gerentes de projeto, como número de linhas inseridas por colaborador, número de *issues* abertas por tipo e número de *commits* por período. Como foram escolhidas três seções diferentes, foram desenvolvidas 3 aplicações que diferem apenas qual parte da API é acessada e quais

campos são retornados. O código da aplicação encontra-se no Apêndice 1.

A Figura 21 apresenta um exemplo de um dos módulos da aplicação, o módulo de extração de estatísticas sobre linhas de código. Para executar a aplicação, é necessário realizar os seguintes passos:

- Instalar Python versão 2.7;
- Baixar e instalar o pacote de Requests do Python³;
- Baixar e instalar o pacote github3 do Python.

```
1 # coding=utf-8
2 import csv
3 import github3
4
5 git_username = 'wbrum'
6 git_api_token = '*****'
7
8 # csv name
9 csv_name = "git_hub_stats.csv"
10
11 def run_csv():
12     # Exporta estatísticas de um repositório em formato CSV
13     output_csv = csv.writer(open(csv_name, 'wb'), delimiter=',')
14     github = github3.login(username=git_username, token=git_api_token)
15
16     # csv headers
17     headers = [
18         'author',
19         'total',
20         'Start of the Week',
21         'Deletions',
22         'Additions',
23         'Commits',
24     ]
25
26     # escreve header rows
27     output_csv.writerow(headers)
28
29     # baixa estatísticas do repositório e transcreve para csv
30     git_stats = github.repository("gems-uff", "noworkflow").iter_contributor_statistics()
31     for git_stat in git_stats:
32         for git_week in git_stat.alt_weeks:
33             commit = [
34                 git_stat.author,
35                 git_stat.total,
36                 git_week['start of week'],
37                 git_week['deletions'],
38                 git_week['additions'],
39                 git_week['commits'],
40             ]
41             output_csv.writerow(commit)
42
43 if __name__ == '__main__':
44     run_csv()
```

Figura 21 - Aplicação de Extração de Estatísticas sobre linhas de código

O PyNdicators precisa da conta de um usuário do Github e de sua *api token*, gerada pelo próprio Github, com a finalidade de permitir acesso a API da plataforma. Após

³ <https://pypi.python.org/pypi/requests#downloads>

especificar esses dados, o PyNdicators inicia a criação de um arquivo CSV (formato de arquivo texto onde os valores são separados por vírgulas), isto é, o arquivo no qual os dados serão escritos localmente, e realiza login na API utilizando as informações fornecidas.

O próximo passo do PyNdicators é criar os cabeçalhos para cada tipo de dado a ser extraído e em seguida acessa a seção desejada da API através do pacote github3. Nesse passo é necessário especificar o dono do repositório e o repositório. No exemplo apresentado, o dono seria “gems-uff” e o repositório “noworkflow”⁴.

Utilizando como exemplo a aplicação de extração de estatísticas sobre linhas de código, o acesso às estatísticas é realizado com o método “*iter_contributor_statistics*”, que realiza uma iteração pela lista de contribuidores para aquele repositório. Em seguida, são realizadas duas iterações: uma sobre cada elemento retornado na iteração mencionada anteriormente e outra sobre as semanas de cada estatística a fim de separar o dicionário de dados *alt_weeks* por semana. Com os dados retornados, criamos uma coleção “commit”, que será representará cada linha do arquivo CSV a ser gerado. Assim, a linha do arquivo CVS será formada por autor, total de *commits* daquele autor e um dicionário que contém a data de início da semana, número de linhas removidas e adicionadas, e o número de *commits* realizados. A cada coleção commit formada, ela é escrita no arquivo CSV até que todas as iterações de *git_stat* terminem.

A Figura 22 apresenta um exemplo de conteúdo de arquivo CSV gerado.

⁴ Foi utilizado para o estudo de caso um repositório real utilizado por um grupo de pesquisadores da Universidade Federal Fluminense. O repositório se encontra no seguinte endereço: <https://github.com/gems-uff/noworkflow>.

author	total	Start of the Week	Deletions	Additions	Commits
JoaoFelipe	268	21/05/2016 21:00	0	0	0
leomurta	81	21/05/2016 21:00	0	0	0
braganholo	35	21/05/2016 21:00	0	0	0
remram44	1	21/05/2016 21:00	0	0	0
benmarwick	1	21/05/2016 21:00	0	0	0
JoaoFelipe	268	14/05/2016 21:00	0	0	0
leomurta	81	14/05/2016 21:00	0	4	1
braganholo	35	14/05/2016 21:00	0	0	0
remram44	1	14/05/2016 21:00	0	0	0
benmarwick	1	14/05/2016 21:00	0	0	0
JoaoFelipe	268	07/05/2016 21:00	0	0	0
leomurta	81	07/05/2016 21:00	0	0	0
braganholo	35	07/05/2016 21:00	0	0	0
remram44	1	07/05/2016 21:00	0	0	0
benmarwick	1	07/05/2016 21:00	0	0	0
JoaoFelipe	268	30/04/2016 21:00	0	0	0
leomurta	81	30/04/2016 21:00	0	0	0
braganholo	35	30/04/2016 21:00	0	0	0
remram44	1	30/04/2016 21:00	0	0	0
benmarwick	1	30/04/2016 21:00	0	0	0
JoaoFelipe	268	23/04/2016 21:00	5	6	2
leomurta	81	23/04/2016 21:00	0	0	0
braganholo	35	23/04/2016 21:00	0	0	0
remram44	1	23/04/2016 21:00	0	0	0
benmarwick	1	23/04/2016 21:00	0	0	0
JoaoFelipe	268	16/04/2016 21:00	1	587	2

Figura 22 - Exemplo do produto final da extração de estatísticas sobre linhas de código

A fim de dar suporte aos usuários dos indicadores, uma planilha eletrônica em Excel deve ser usada para a construção dos gráficos dos indicadores a partir das bases geradas pelo PyNdicators. Após a geração dos três arquivos CSV, o gerente do projeto deve atualizar a planilha eletrônica e selecionar as bases geradas para atualizar automaticamente todos os gráficos elaborados.

3.2 Formulação dos Indicadores Complementares

Como um dos objetivos do trabalho é apresentar indicadores que possam ser complementares àqueles do Github, foram realizadas análises, com base na API do Github, nos dados fornecidos e nos indicadores já existentes do Github, sobre que tipos

de indicadores poderiam ser formulados e como eles apoiariam gerentes e equipes de projeto.

Os indicadores elaborados foram:

1. RIC – Rótulos de *Issues* Mais Comuns
2. IAF - *Issues* Abertas x *Issues* Fechadas
3. NIA - Número de *Issues* Com x Sem Responsável
4. TFI - Tempo para fechamento de *issues*
5. NIC - Número de *Issues* criadas por mês / ano
6. CMA – *Commits* por mês / ano
7. IEA – Inclusões x Exclusões por Autor
8. PCA – Porcentagem de *commits* por Autor
9. RCM – Ranking de Colaborador por Mês

Os indicadores foram construídos a fim de melhor atender as principais necessidades de um gerente de projetos, levando em consideração o viés *open source* dos projetos do Github. Conforme mencionado na Seção 2.4, apesar de os indicadores do Github apresentarem informações relevantes de desempenho, eles apresentam dados apenas de *commits* e linhas de código alteradas, isto é, não atendem a funcionalidade de *issues*.

Os seguintes indicadores foram construídos a partir da base de *issues* gerada pelo PyNdicators:

- O Rótulo de *Issues* Mais Comuns (RIC) foi construído a partir da ideia que é importante para um gerente de projetos ter a visão sobre os tipos de *issues* que estão mais presentes durante o desenvolvimento do projeto. Apesar de os tipos serem diferentes para cada projeto, a visão ainda é relevante nos contextos específicos de cada desenvolvimento.
- O *Issues* Abertas x *Issues* Fechadas (IAF) permite ao gerente do projeto visualizar a relação entre a quantidade de *issues* que estão sendo fechadas e quantas ainda estão em aberto, isto é, a equipe ainda não conseguiu resolver. É uma visão importante, pois o gerente consegue observar se a equipe está sendo sobrecarregada de *issues*, ou até de um tipo de *issue* em particular, e tomar as decisões necessárias para reverter a situação.
- O Número de *Issues* Com x Sem Responsável (NIA) é um indicador relacionado ao controle da equipe do projeto e do gestor. A partir dele é

possível visualizar o número de *issues* que estão abertas e fechadas com ou sem o campo *assignee* (responsável) preenchido. O não preenchimento desse campo pode indicar, por exemplo, uma falha no processo de preenchimento das *issues* por parte da equipe ou do gestor. Isso pode levar a tarefas ociosas, pois nenhum colaborador está ciente dela ou quem deve cumpri-la, algo que acaba impactando no andamento do projeto.

- O Tempo de fechamento de *Issues* (TFI) apresenta o relacionamento entre a quantidade de dias para fechamento de cada issue e a quantidade de *issues* que foram fechadas naquele tempo. Esse indicador permite ao gerente de projeto montar objetivos relacionados a um tempo limite de fechamento, ou seja, uma data aceitável para fechamento da issue até ela se tornar uma preocupação para a equipe e o projeto, dependendo da sua gravidade.
- O Número de *Issues* Criadas por mês / ano (NIC) é um indicador que permite ao gerente de projetos visualizar a quantidade de *issues* e tipos de *issues* criados por mês e ano. Esse indicador permite medir se em algum mês em particular houve alguma disparidade em relação ao número de *issues* criadas, tanto com um número acima do normal quanto abaixo. Por exemplo, se uma issue do tipo Defeito se destacar em alguma seção do gráfico, o gerente precisa se reunir com a equipe e entender o porquê de isso ter acontecido.

O próximo indicador foi criado a partir da base de *commits* gerada pelo PyNdicators:

- O indicador *Commits* por mês / ano (CMA) apresenta uma visão complementar ao NIC, só que voltada para o desempenho dos colaboradores. O número de *commits* por cada colaborador é apresentado em cada mês, permitindo o gerente a avaliar que colaboradores mais realizaram *commits* em cada mês e relacionar essa informação às tarefas atribuídas a cada colaborador em dado período de tempo.

Os últimos três indicadores foram elaborados a partir da base de estatísticas da API do Github:

- O Inclusões x Exclusões por Autor (IEA) é o indicador responsável pelo acompanhamento do trabalho realizado por cada colaborador através de linhas de código incluídas e excluídas. Ele funciona de forma complementar

ao CMA em termos de medir o desempenho do colaborador, pois um colaborador com um número alto de *commits* no projeto tende a, geralmente, possuir um número alto de linhas incluídas e excluídas de código. Um número alto de linhas excluídas de código por um colaborador pode, por exemplo, indicar que este colaborador trabalhou com muita otimização de código legado e removeu trechos de código que não são mais necessários. Ess indicador é similar ao já disponível no Github, porém possui uma forma de visualização com dados mais concretos para o gerente de projetos devido à forma como o gráfico é construído e exibido.

- O Porcentagem de *Commits* por Autor (PCA) permite ao gerente de projeto visualizar a participação de cada colaborador baseado no número de *commits* por cada autor. Ele se relaciona diretamente com o CMA e com o Ranking de Colaborador por Mês (RCM) e, como o CMA, indiretamente com o IEA. Um colaborador com uma porcentagem elevada nesse indicador pode indicar que muitas *issues* estão sendo atribuídas a ele e a redistribuição de tarefas para o resto da equipe seja necessária.
- O Ranking de Colaborador por Mês (RCM) é um gráfico de suma importância para um gerente de projetos, pois permite a visualização de qual colaborador, através do número de *commits*, está sendo mais ativo na execução do projeto. Além disso, permite o gerente visualizar se, por exemplo, há poucos *commits* no total dentro de um mês, que pode ser uma reflexão da dificuldade das tarefas de cada colaborador, ou se um colaborador em específico está sendo pouco ativo e se reunir com ele para entender o motivo do desempenho e trabalhar melhor a distribuição de atividades, se for o caso.

3.3 Aplicação do Template AMP.br

A medição de software, como explicado anteriormente, é fundamental para o ambiente organizacional, independentemente do tamanho do projeto, e para a visibilidade do andamento, do progresso e dos resultados de um projeto.

Como cada projeto possui seu próprio cenário, objetivos, especificações e limitações, os indicadores variam, de acordo com as características do projeto e o contexto no qual se insere. Por isso, selecionar o conjunto de informações adequadas

para o projeto é uma tarefa importante para estes indicadores possuírem relevância, buscando facilitar a seleção e os seus usos no projeto. O projeto apresentado nesse trabalho de conclusão foi elaborado considerando um projeto de desenvolvimento *open source* que utiliza a ferramenta de versionamento GitHub.

O template AMP.br (Apoio a Medição de Projetos de Software – Uma abordagem baseada em relacionamentos), desenvolvido por BONELLI (2014), permite registrar os indicadores em fichas de registro de indicadores contendo alguns elementos base para facilitar o registro e também a análise do indicador posteriormente. O conjunto de indicadores é representado no mapa de Objetivos, de Indicadores e no mapa de MR-MPS-SW. O template possui três mapas principais para o registro de categorias e relacionamentos dos indicadores com o intuito de auxiliar a percepção do relacionamento entre os indicadores, os objetivos e os níveis/processos do MR-MPS-SW e, como consequência, apoiar o gerente de projeto na tomada de decisão e elaboração do seu plano de medição.

3.3.1 Construção do Mapa de Objetivos

O mapa de objetivos contém todos os objetivos reunidos, isto é, objetivos estratégicos e objetivos de medição relacionados com o elemento necessidade de informação, considerando que estes objetivos tanto quanto os indicadores foram elaborados considerando projetos de desenvolvimento *open source* no GitHub.

O elemento mais externo do mapa, necessidade de informação, descreve quais questionamentos serão respondidos para apoiar a tomada de decisão, os objetivos de medição descrevem as ações que são planejadas e realizadas no campo da medição e o elemento mais central do mapa, os objetivos estratégicos, descrevem as ações que são planejadas e realizadas no campo estratégico. Os objetivos estratégicos elaborados foram genéricos de tal forma que pudessem ser utilizados por uma gama maior de organizações. Todavia, as organizações devem adaptar os objetivos estratégicos para os seus próprios a fim de utilizar os indicadores.

A tabela com as necessidades de informação, o seu objetivo de medição relacionado e o seu objetivo estratégico relacionado é apresentada na Tabela 3.

Tabela 3 - Relacionamento entre os Objetivos Estratégicos, de Medição e as Necessidades de Informação

Objetivo estratégico	Objetivo de medição	Necessidade de informação
Aumentar a eficácia da gestão do projeto	Monitorar a eficácia dos colaboradores em fechar <i>issues</i> no projeto.	O número de <i>issues</i> abertas está muito grande para a equipe atual?
	Monitorar os tipos de tarefas mais comuns no desenvolvimento do projeto	Quais tipos de <i>issues</i> estão sendo mais abertas no Projeto?
	Monitorar o número de <i>issues</i> que estão sem responsável	O número de <i>issues</i> sem um responsável está muito alto?
Aumentar a eficácia da gestão da produtividade	Monitorar as conclusões de <i>issues</i> no projeto	Qual a quantidade de tempo em que as <i>issues</i> são fechadas?
Aumentar a eficácia da gestão do projeto	Monitorar a criação de <i>issues</i> no projeto	Qual o número de <i>issues</i> criadas por mês e ano?
Aumentar a eficácia da gestão do projeto e Aumentar a eficácia da gestão do Código fonte	Monitorar os <i>commits</i> de código fonte no projeto	Qual o número de <i>commits</i> de código fonte realizados por mês e ano?
Aumentar a eficácia da gestão do Código fonte	Monitorar o desempenho dos colaboradores	Quais colaboradores estão contribuindo mais para o projeto?
		Qual colaborador está realizando mais <i>commits</i> de código fonte?
		Qual colaborador está realizando mais <i>commits</i> de código fonte em um dado mês

A relação entre a Necessidade de informação e os indicadores elaborados está apresentada na Tabela 4:

Table 4 - Relacionamento entre necessidades de informação e indicadores

Necessidade de informação	Indicador
Quais tipos de <i>Issues</i> estão sendo mais abertas no Projeto?	RIC
O número de <i>issues</i> abertas está muito grande para a equipe atual?	IAF
O número de <i>Issues</i> sem um responsável está muito alto?	NIA
Qual a quantidade de tempo em que as <i>issues</i> são fechadas?	TFI
Qual o número de <i>issues</i> criadas por mês e ano?	NIC
Qual o número de <i>commits</i> de código fonte realizados por mês e ano?	CMA
Quais colaboradores estão contribuindo mais para o projeto?	IEA
Qual colaborador está realizando mais <i>commits</i> de código fonte?	PCA
Qual colaborador está realizando mais <i>commits</i> de código fonte em um dado mês	RCM

A representação gráfica da relação entre os elementos desta tabela (necessidade de informação, objetivos de medição e objetivos estratégicos) está representada no mapa de objetivos (Figura 23).

3.3.2 Construção do Mapa de Indicadores

O mapa de indicadores (Figura 24) é o artefato responsável pelos relacionamentos entre os indicadores, apresentando todos os indicadores e os seus

relacionamentos através dos procedimentos de análise baseado em relacionamentos preenchido nas fichas de indicadores.

Com o uso da notação apresentada, retas conectam indicadores demonstrando que estes se relacionam e podem ser analisados e interpretados em conjunto para auxiliar na tomada de decisão. Além disso, graficamente é exibido o elemento categoria, baseado no PSM (MCGARRY *et al.*, 2002). No Apêndice 2 estão, além dos mapas elaborados, as Fichas dos Indicadores contendo as justificativas dos relacionamentos entre os indicadores.

3.3.3 Construção do Mapa do MR-MPS-SW

O mapa de MR-MPS-SW (Tabela 5) permite identificar os indicadores, dividindo-os em sua sigla e seu nome por extenso. Em relação ao mapeamento MR-MPS-SW temos os níveis em que os indicadores estão adequados, e estes subdivididos nos processos selecionados. Os indicadores deste trabalho estão relacionados ao nível G e F do MR-MPS-SW, tendo como processos Gerência de Projetos, para o nível G, e Gerência de Configuração, para o nível F.

Table 5 - Mapa do MR-MPS-SW

INDICADORES DE PATTERN		MR-MPS-SW	
		G	F
SIGLA	NOME	GPR	GCO
RIC	RÓTULO DE ISSUES MAIS COMUNS	X	
IAF	ISSUES ABERTAS X FECHADAS	X	
NIA	NÚMERO DE ISSUES COM X SEM RESPONSÁVEL	X	
TFI	TEMPO PARA FECHAMENTO DE ISSUES	X	
NIC	NÚMERO DE ISSUES CRIADOS POR MÊS / ANO	X	
CMA	COMMITTS POR MÊS / ANO		X
IEA	INCLUSÕES X EXCLUSÕES POR AUTOR		X
PCA	PORCENTAGEM DE COMMITTS POR AUTOR		X
RCM	RANKING DE COLABORADOR POR MÊS		X

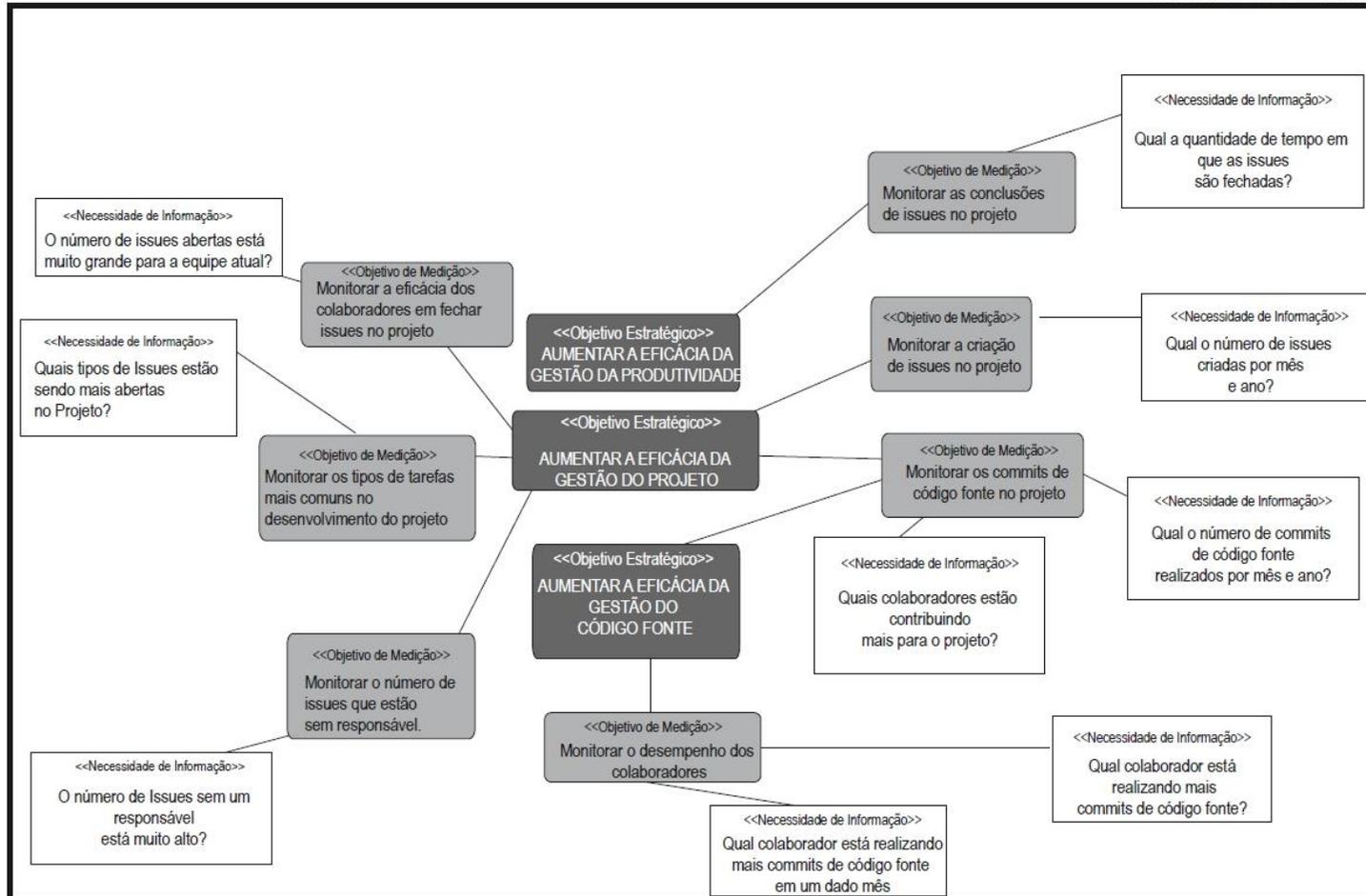


Figura 23 - Mapa de objetivos

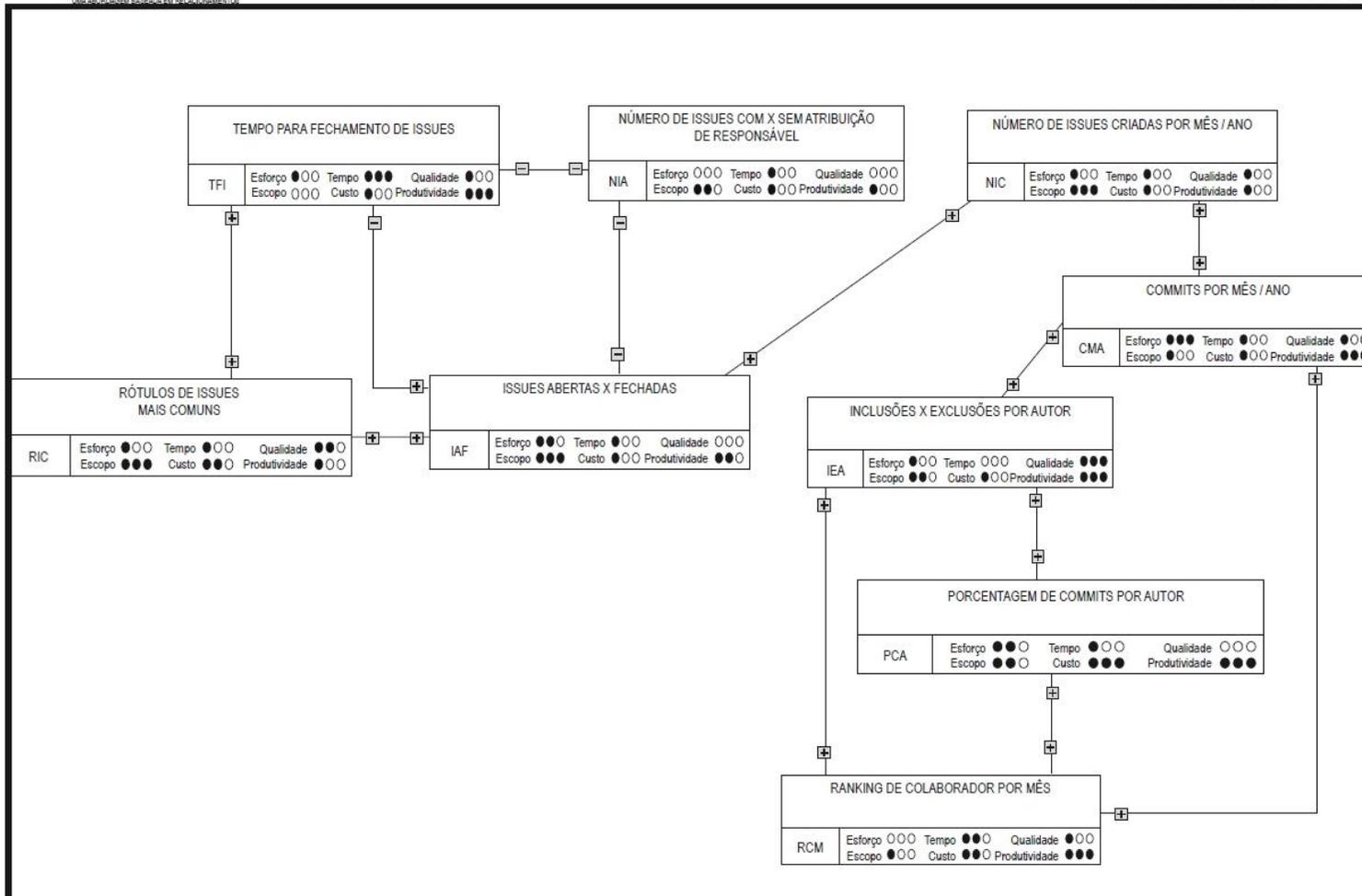


Figura 24 - Mapa de Indicadores

3.3.4 Construção da Ficha de Indicador

As fichas, um dos três componentes principais do template, permite descrever um indicador em apenas uma página com objetivo de simplificar o acesso as informações. As fichas são divididas em áreas, antes já explicadas, que são:

- a) Indicador
- b) Necessidade da informação
- c) Categoria
- d) Objetivos
- e) Nível MR-MPS-SW
- f) Representação Gráfica
- g) Procedimento de medição e análise
- h) Procedimento de análise baseado em critérios
- i) Procedimento de análise baseado em relacionamentos

Com a intenção de utilizar o modelo de forma a automatizar os procedimentos de análise de projetos em repositórios de versionamento, como o GitHub, abordaremos como utilizamos este template para registrar indicadores para a gestão do projeto, tendo como fonte os dados obtidos na plataforma do GitHub via PyNdicators.

Como dito anteriormente, as fichas geradas se referem aos seguintes indicadores:

1. RIC – Rótulos de *Issues* Mais Comuns
2. IAF - *Issues* Abertas x *Issues* Fechadas
3. NIA - Número de *Issues* Com x Sem Responsável
4. TFI - Tempo para fechamento de *issues*
5. NIC - Número de *Issues* criadas por mês / ano
6. CMA – *Commits* por mês / ano
7. IEA – Inclusões x Exclusões por Autor
8. PCA – Porcentagem de *commits* por Autor
9. RCM – Ranking de Colaborador por Mês

Para apresentar o preenchimento do template e como ele pode auxiliar os projetos baseados em repositórios do tipo GitHub, usaremos como exemplo os indicadores: RIC– Rótulos de *Issues* Mais Comuns (Figura 25) e o IAF - *Issues* Abertas x *Issues* Fechadas (Figura 26). Estas e as demais fichas do template AMP.br estão disponíveis no Anexo 2 deste trabalho.

Objetivos

Objetivo Estratégico: Aumentar a eficácia da gestão do projeto

Objetivo de Medição: Monitorar os tipos de tarefas mais comuns no desenvolvimento do projeto

Tipo: controle, qualidade, desempenho

MR-MPS-SW
 NÍVEL: G
 PROC: GPR

Representação Gráfica do Indicador

Contagem de labels

Label	closed	open
[]	3	3
[bug]	29	5
[enhancement, related work]	1	1
[enhancement]	32	27
[Library, related work]	2	2
[Library]	3	3
[related work]	0	8

state: closed (blue), open (red)

labels: []

Procedimento de Medição e Análise

MEDICÃO

INÍCIO: Em qualquer fase do projeto, respeitando a frequência de execução do indicador

Medida: RIC

Coletar dados sobre os rótulos e estados das Issues na área de Issues do Github através do PyNdicators

Atualizar dados na Planilha de Medição do Indicador

Planilha de Medição Atualizada

ANÁLISE

FIM

Executar o Procedimento de Análise Baseado em Relacionamentos

Avaliar gráfico gerado e executar o Procedimento de Análise Baseado em Critérios disponibilizado na Ficha de Descrição do Indicador

Geração automática do gráfico para a visualização das Labels de Issues mais comuns na planilha de medição (RIC)

Selecionar dados relevantes para análise. (Especificar período de medição e/ou tipos/estados de issues).

Procedimento de Análise Baseado em Relacionamentos

Indicador RIC

IAB: Issues Abertas x Fechadas

O aumento no indicador de Issues Abertas x Fechadas impacta diretamente em diversos tipos de issues aumentarem devido ao número crescente de issues criadas. Desta forma, o gerente de projeto deve avaliar qual o tipo de issue que está sendo mais criado.

TFI: Tempo Médio de Fechamento de Issues

O gerente do projeto pode avaliar quais tipos de issues estão tomando mais ou menos tempo para serem concluídos e verificar com a equipe as possíveis complexidades desse tipo de issue. Um número crescente issues de um tipo específico também aumenta no número de issues que são eventualmente fechadas e o seu tempo de fechamento.

Medida Base: RIC: RÓTULOS ATRIBUÍDOS A CADA ISSUE

Periodicidade: MENSAL

Responsáveis: Coleta: GERENTE DO PROJETO, Análise: GERENTE DO PROJETO

Procedimento de Análise Baseado em Critérios			
ID	Premissa	Possível Conclusão	Possíveis Ações
1	Número de Issues Abertas é maior que Issues Fechadas em 10 unidades	Isso pode ocorrer devido a algum problema na execução do procedimento de cadastro de Issues, falta de experiência, número baixo de colaboradores na equipe, etc.	A equipe deve identificar quais atividades estão criando essas demandas de aberturas de issues e como resolver de maneira mais eficiente os problemas encontrados
2	Elementos no eixo X sem nome	Isso pode ocorrer devido a algum problema na execução procedimento de cadastro de Issues ou falta de experiência pelos criadores de issues	A equipe deve se reunir e identificar que issues estão sem tipo e definir em conjunto o tipo mais apropriado para cada uma.
3	Um tipo de issue apresenta 50% ou mais do total de tipos de issue	Um tipo de issue recorrente pode indicar um ciclo de melhorias na aplicação, devido a requisições constantes de novas funcionalidades, bem como pode indicar a presença de erros na aplicação	Ao identificar o tipo de Issue recorrente, a equipe deve-se reunir e compreender a razão daquele tipo aparecer muitas vezes. Se for do tipo bug, por exemplo, a equipe deve tomar mais atenção com a fase de testes durante o desenvolvimento
4	Número de Issues Fechadas é maior que Issues Abertas em 10 unidades	Os colaboradores do projeto estão concluindo as Issues corretamente e resolvendo os possíveis problemas apresentados.	Não há ação necessária

PÁG. 11/11

Figura 25 - Ficha do Indicador RIC

41

A “necessidade de informação” é a parte da ficha que descreve qual questionamento será respondido para apoiar a tomada de decisão, para isto podemos inferir que na Ficha 1, do indicador RIC – Rótulos de *Issues* Mais Comuns, há a necessidade de saber “Quais tipos de *Issues* estão sendo mais abertas no Projeto?”, pois é relevante notar qual o tipo de *issue* que está sendo aberta e assim notar quais as tarefas mais comuns ou também se há um número de bugs/defeitos que demanda maior atenção.

Já na Ficha 2, IAF - *Issues* Abertas x *Issues* Fechadas, há a necessidade de saber “O número de *issues* abertas está muito grande para a equipe atual?”, pois também é relevante para o gerente monitorar a eficácia dos colaboradores em fechar as *issues* no projeto.

Para o elemento “Categoria”, seguindo o template, e este baseando-se no PSM, as categorias que o indicador se relaciona e principalmente o grau (o preenchimento – relacionado à influência e ao impacto nesta categoria) para os indicadores seriam os indicados na Figura 27 (para o indicador RIC) e Figura 28 (para o indicador IAF):

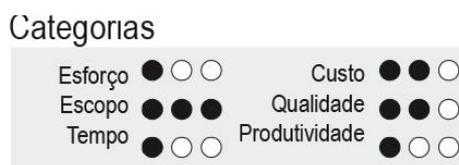


Figura 27 - Categorias do Indicador RIC

O RIC possui relação de influência indireta para a categoria tempo, esforço e produtividade, influência direta na categoria qualidade e custo e impacto direto em escopo.

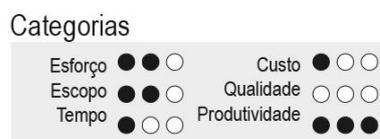


Figura 28 - Categorias do Indicador IAF

Por outro lado, o IAF possui relação de influência indireta para as categorias tempo e custo, impacto direto na categoria produtividade, influência direta nas categorias esforço e escopo e não há influência em qualidade.

Para o elemento “Objetivos”, o qual é dividido em objetivos estratégicos e objetivos de medição, há a descrição das ações que são planejadas e realizadas, um no campo estratégico outro no campo da medição.

Para RIC, o objetivo estratégico é “Aumentar a eficácia da gestão”, obtido através da análise e observação de quais os rótulos mais comuns para as *issues*, o que reflete também no objetivo de medição “Monitorar os tipos de tarefas mais comuns no desenvolvimento do projeto”.

Para IAF, o objetivo estratégico é “Aumentar a eficácia da gestão”, obtido através da análise e observação da quantidade de *issues* abertas e fechadas divididas pelos rótulos das *issues*, o que reflete também no objetivo de medição “Monitorar a eficácia dos colaboradores em fechar *issues* no projeto”.

Para o elemento “Nível MR-MPS-SW”, tem-se a finalidade de apresentar para o gerente de projetos o nível MR-MPS-SW, que possui resultados esperados os quais podem ser apoiados com o uso do indicador que está representado na ficha. Para ambos os indicadores, o processo do MR-MPS-SW associado à Gerência de Projetos (GPR) presente no nível G de maturidade (Figura 29). Assim, indica-se que apoio dos indicadores RIC e IAF neste projeto é condizente ao nível G do MR-MPS-SW e relacionados ao GPR (Gerência de Projetos).

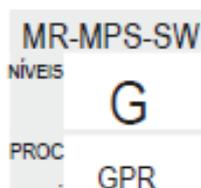


Figura 29 - Nível MR-MPS-SW dos indicadores RIC e IAF

Para o elemento “Representação Gráfica”, a área do template referente à representação visual/gráfica do indicador apresenta a plotagem dos dados obtidos via aplicação representados de forma a facilitar a visualização e análise.

A Figura 30 apresenta o gráfico para o indicador RIC. Nesta representação gráfica temos os rótulos das *issues* mais comuns no projeto, divididos em seus *status* (*issues* abertas e fechadas).

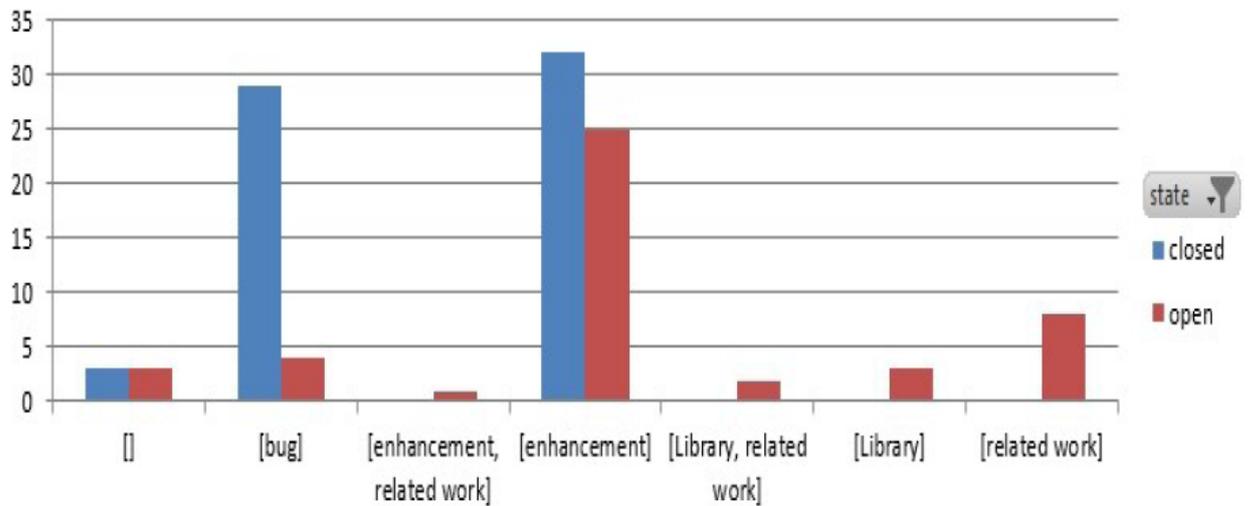


Figura 30 - Gráfico do Indicador RIC

A Figura 31 apresenta o gráfico para o indicador IAF. Nesta representação gráfica temos a quantidade consolidada de *issues* abertas e fechadas divididas internamente (cores e legenda) pelos rótulos das *issues*.

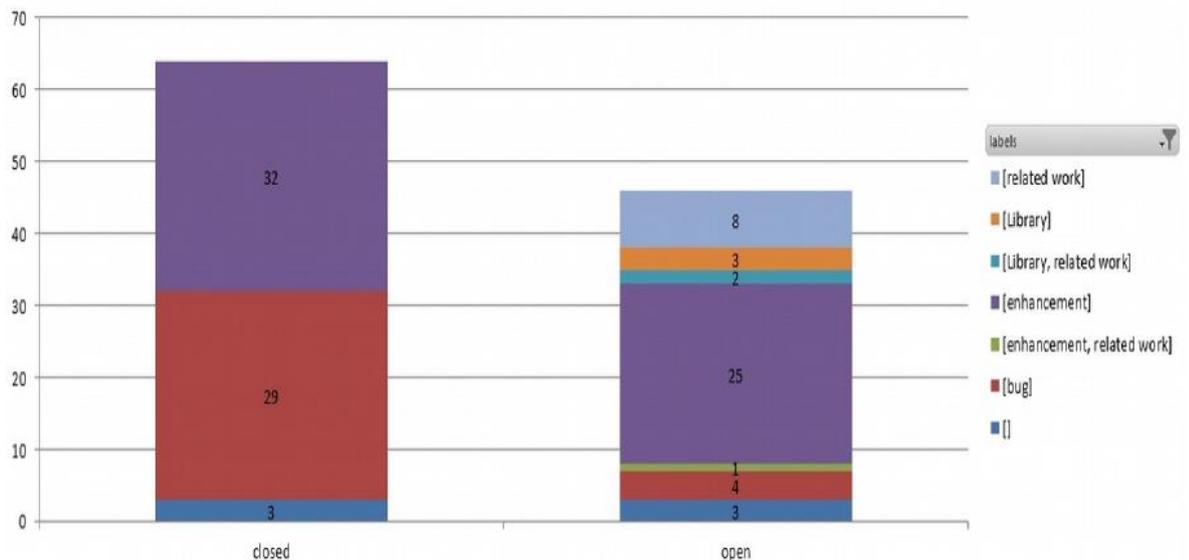


Figura 31 - Gráfico do Indicador IAF

Na seção “Procedimento de medição e análise”, que representa o fluxo de atividades necessárias para que a medição e análise ocorram, é observado o processo da coleta dos dados, a atualização da planilha de medição e, por fim, a análise dos dados

A Figura 32 apresenta o procedimento de medição e análise para o indicador RIC.

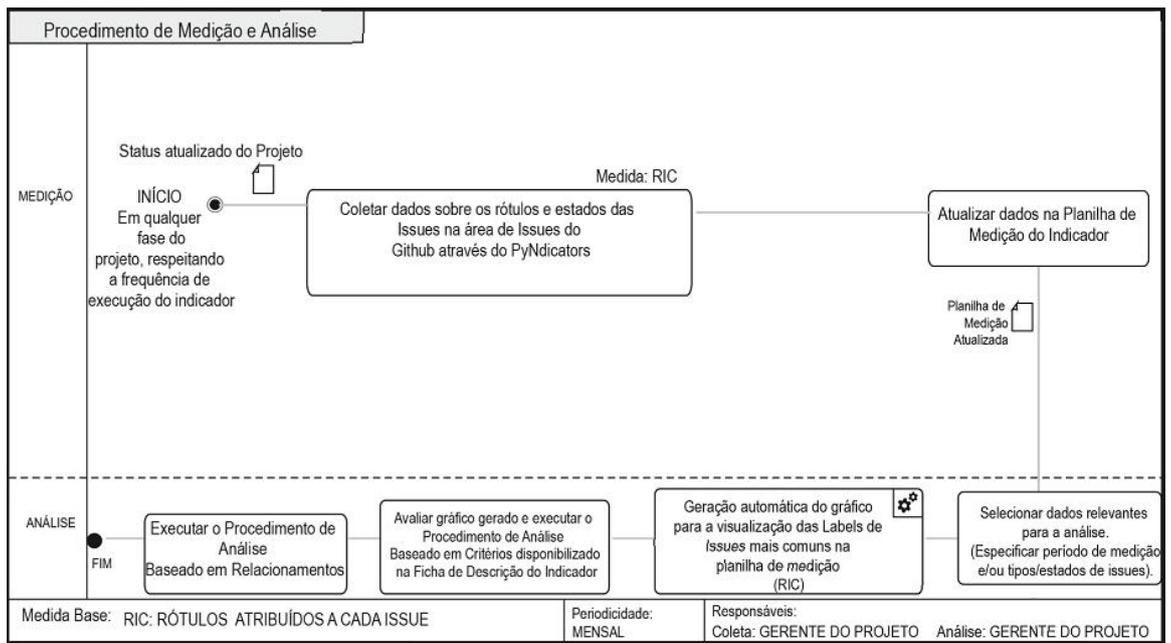


Figura 32 - Procedimento de Medição e Análise do indicador RIC

A Figura 33 apresenta o procedimento de medição e análise para o indicador IAF.

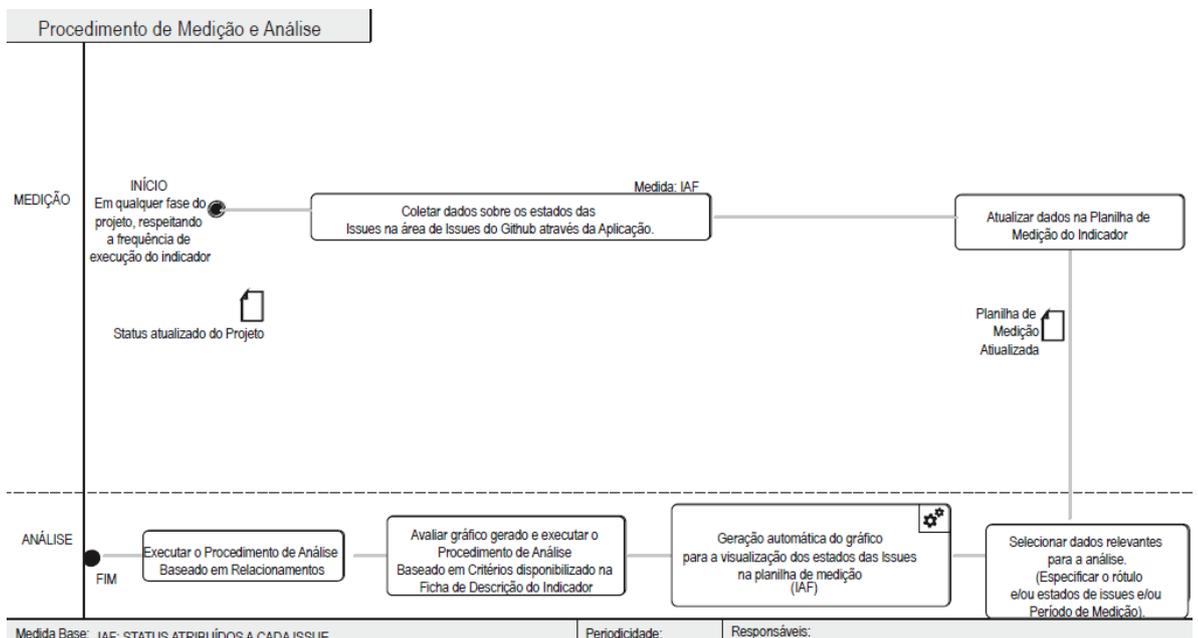


Figura 33 - Procedimento de Medição e Análise do indicador IAF

Pode-se perceber que o início do fluxo para todos os indicadores, na parte da coleta de dados, é muito semelhante. O que altera são as medidas e quais os dados são coletados pela aplicação que exportará os dados do Github.

A Figura 34 apresenta o procedimento de medição para o indicador RIC:

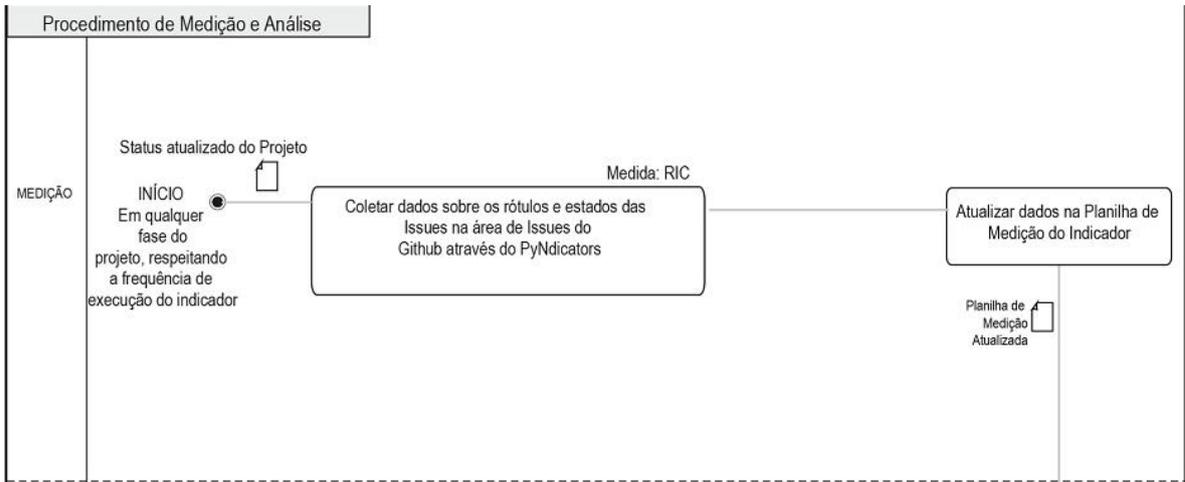


Figura 34 - Procedimento de Medição - RIC

A Figura 35 apresenta o procedimento de análise do indicador RIC:

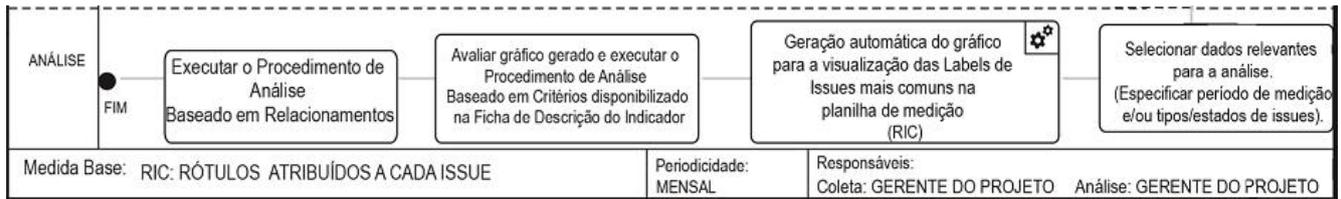


Figura 35 - Procedimento de Análise - RIC

A seleção dos dados é feita, especificando quais os campos e detalhes que se fazem necessário para constar no indicador. O gráfico é gerado automaticamente para visualização e assim possibilitar o uso do indicador para o template e principalmente para análise posterior pelo gerente.

Para o elemento “Procedimento de análise baseado em critérios” (h), definimos premissas, possíveis conclusões e, ações a serem tomadas visto essa conclusão.

A Figura 36 apresenta o procedimento de análise baseado em critérios do indicador RIC:

Procedimento de Análise Baseado em Critérios			
ID	Premissa	Possível Conclusão	Possíveis Ações
1	Número de Issues Abertas é maior que Issues Fechadas em 10 unidades	Isso pode ocorrer devido a algum problema na execução do procedimento de cadastro de Issues, falta de experiência, número baixo de colaboradores na equipe, etc.	A equipe deve identificar quais atividades estão criando essas demandas de aberturas de issues e como resolver de maneira mais eficiente os problemas encontrados
2	Elementos no eixo X sem nome	Isso pode ocorrer devido a algum problema na execução do procedimento de cadastro de Issues ou falta de experiência pelos criadores de issues	A equipe deve se reunir e identificar que issues estão sem tipo e definir em conjunto o tipo mais apropriado para cada uma.
3	Um tipo de issue apresenta 50% ou mais do total de tipos de issue	Um tipo de issue recorrente pode indicar um ciclo de melhorias na aplicação, devido a requisições constantes de novas funcionalidades, bem como pode indicar a presença de erros na aplicação	Ao identificar o tipo de Issue recorrente, a equipe deve-se reunir e compreender a razão daquele tipo aparecer muitas vezes. Se for do tipo bug, por exemplo, a equipe deve tomar mais atenção com a fase de testes durante o desenvolvimento
4	Número de Issues Fechadas é maior que Issues Abertas em 10 unidades	Os colaboradores do projeto estão concluindo as Issues corretamente e resolvendo os possíveis problemas apresentados.	Não há ação necessária

Figura 36 - Procedimento de Análise Baseado em Critérios - RIC

A Figura 37 apresenta o procedimento de análise baseado em critérios do indicador IAF:

Procedimento de Análise Baseado em Critérios			
ID	Premissa	Possível Conclusão	Possíveis Ações
1	Número de Issues Abertas é muito maior que o número de issues fechadas.	Isso pode ocorrer devido a algum problema na execução de um determinado processo, falta de experiência, número baixo de colaboradores na equipe, etc.	A equipe deve identificar quem são os responsáveis pela resolução das issues, identificar os problemas e solucioná-los
2	Um tipo de Issue está se destacando em uma das colunas	Um tipo de issue recorrente pode indicar um ciclo de melhorias na aplicação, devido a requisições constantes de novas funcionalidades, bem como pode indicar a presença de erros na aplicação	Ao identificar o tipo de Issue recorrente, a equipe deve-se reunir e compreender a razão daquele tipo aparecer muitas vezes. Se for do tipo bug, a equipe deve tomar mais atenção com a fase de testes durante o desenvolvimento
3	Issues sem tipo no gráfico	Isso pode ocorrer devido a algum problema na execução de um determinado processo ou falta de experiência pelos criadores de issues	A equipe deve se reunir e identificar que issues estão sem tipo e definir em conjunto o tipo mais apropriado para cada uma.
4	Número de Issues Fechadas é muito maior que o número de issues Abertas.	Os colaboradores do projeto estão concluindo as Issues corretamente e resolvendo os possíveis problemas apresentados.	Não há ação necessária

Figura 37 - Procedimento de Análise Baseado em Critérios - IAF

O procedimento de análise baseado em critérios foi preenchido observando as premissas que poderiam ser observadas na análise dos indicadores para os projetos. Ela garante uma visão inicial de uma observação já mostrando a possível conclusão no caso desta premissa ser observada e qual a tomada de decisão que poderá ser realizada (possíveis ações).

O elemento “Procedimento de análise baseado em relacionamentos” apresenta o procedimento baseado no relacionamento indicadores, isto é, como um indicador influencia, positiva ou negativamente, outro indicador.

A Figura 38 apresenta o procedimento de análise baseado em relacionamentos do indicador RIC:

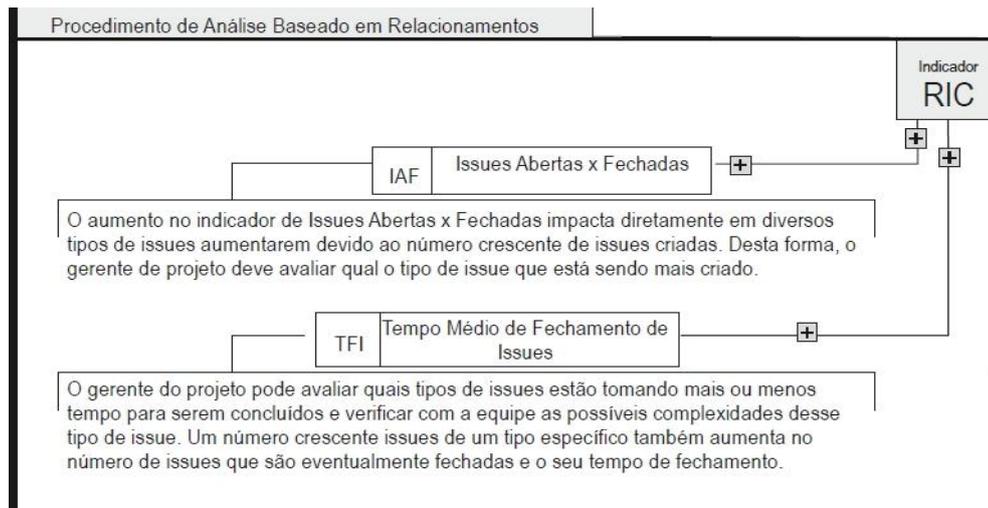


Figura 38 - Procedimento de Análise Baseado em Relacionamentos - RIC

Na Figura 38, pode-se ver que o indicador RIC está relacionado com mais dois indicadores, IAF - *Issues* abertas e fechadas e TFI - Tempo médio de fechamento de *issues*. Esses dois indicadores possuem como interseção o fato de estarem relacionando os rótulos das *issues*. Podendo assim, em conjunto, gerar uma análise mais detalhada e auxiliar o gerente do projeto na tomada de decisão.

A Figura 39 apresenta o procedimento de análise baseado em relacionamentos do indicador IAF:

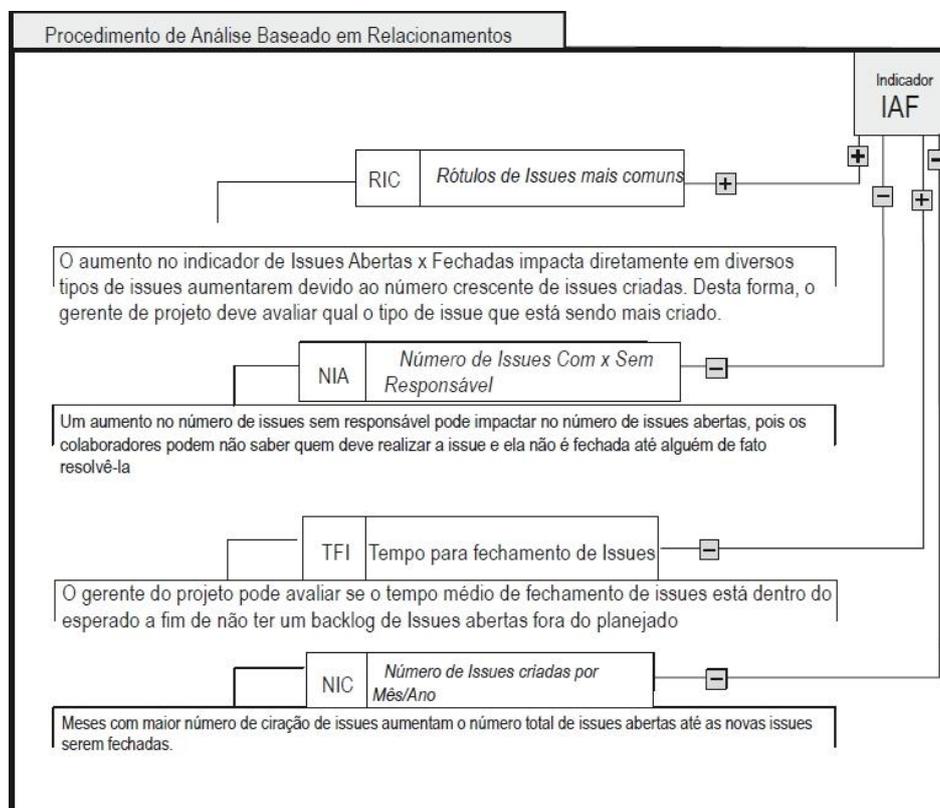


Figura 39 - Procedimento de Análise Baseado em Relacionamentos - IAF

Na figura 39, pode-se ver que o indicador IAF se relaciona com mais quatro indicadores, pois todos abordam em sua análise o *status* das *issues*. Sabendo se as *issues* estão fechadas ou abertas, para todos os quatro indicadores, teremos a opção de em conjunto gerar uma análise mais detalhada e auxiliar o gerente do projeto na tomada de decisão.

3.4 Considerações Finais

Nesse capítulo foram apresentadas a aplicação responsável pela coleta de dados para a construção dos indicadores (PyNdicators) e um exemplo de utilização do template AMP.br em um projeto do Github. O uso do exemplo teve a finalidade de demonstrar como ele é construído a partir das informações coletadas e que é possível ser utilizado para representar indicadores de projetos do Github.

Além disso, espera-se que os indicadores desenvolvidos nesse trabalho possam ser utilizados como uma base na elaboração de planos de medição para gerentes de projeto que utilizam o Github. A utilização da aplicação para coletar os dados da API e o uso de uma planilha eletrônica para atualização dos gráficos dos indicadores são essenciais para o preenchimento das Fichas de Registro de Indicadores. A atualização dos dados foi feita de tal maneira que o gerente de projeto precisa apenas executar a aplicação e atualizar os dados da planilha eletrônica para manter os indicadores atualizados.

No próximo capítulo será abordado o estudo de caso dos indicadores criados no âmbito desse trabalho. O estudo de caso consistiu na entrevista de um dos principais desenvolvedores de um projeto hospedado no Github e de um gerente de projetos de software com experiência na elaboração e análise de indicadores de projetos de software.

4. Avaliação dos Indicadores Construídos

Neste capítulo é abordado um estudo de caso realizado com um projeto no Github, utilizando o PyNdicators para coleta de dados do projeto e aplicando o AMP.br para a construção dos mapas e fichas de indicadores. Além disso, foi entrevistado um colaborador do projeto e foi apresentado o produto gerado e coletado feedback sobre o template. Com a avaliação, foi possível determinar se o uso dos indicadores padronizados a partir do AMP.br supre a necessidade de acompanhamento do projeto tão bem ou melhor que os indicadores já existentes no Github. Também foi entrevistado um gerente de projetos com experiência em projetos e indicadores de software a fim de avaliar os indicadores elaborados e sua aplicação no AMP.br.

4.1 Introdução

Neste capítulo serão abordados os resultados de um estudo de caso realizado em um projeto de desenvolvimento de software hospedado no Github com a finalidade de coletar *feedback* sobre os indicadores criados e a forma de apresentação dos mesmos através do AMP.br.

Através da entrevista com colaboradores de projetos do Github é possível verificar a viabilidade do uso do AMP.br como opção de representação dos indicadores do projeto. Além disso, a entrevista permite que os colaboradores sugiram possíveis mudanças no *template* a fim de facilitar o uso pela equipe e permitir novas visões dos indicadores.

A avaliação do *template* buscou apresentar, através de uma entrevista, a opinião dos colaboradores de um projeto sobre os seguintes tópicos:

- Facilidade de uso do AMP.br;

- Utilidade do AMP.br na elaboração de um plano de medição e avaliação dos resultados;
- Comparação entre os modelos apresentados pelo AMP.br e os disponíveis no Github.

Todos os procedimentos utilizados foram adaptados de (BONELLI , 2014).

4.2 Método de Pesquisa para o Estudo de Caso

As seguintes etapas foram seguidas com o propósito de realizar a avaliação proposta neste trabalho:

1. Elaboração do questionário
2. Identificação de um projeto ativo de desenvolvimento de software no Github;
3. Análise do perfil profissional dos participantes;
4. Procedimento de entrevista;
5. Análise dos resultados da avaliação.

4.2.1 Elaboração do questionário

As questões formuladas para o questionário de avaliação do *template* foram baseadas nos tópicos mencionados na seção 4.1. A Figura 40 representa o *TCLE – Termo de Consentimento e Livre Esclarecimento* que foi utilizado com o propósito de formalizar a participação dos entrevistados no estudo de caso. Logo após ser descrito o objetivo da avaliação ao entrevistado, este documento é apresentado e a assinatura dele representa a autorização para o uso de informações transmitidas.

		TERMO DE CONSENTIMENTO LIVRE E ESCLARECIMENTO (TCLE) Universidade Federal do Estado do Rio de Janeiro, Centro de Ciências Exatas e Tecnologia.	
Pesquisadores Cesar Barbosa e William Brum		Orientadores: Gleison Santos, D.Sc.	
Participante		Nome	
		Email	
Descrição do Estudo			
<p>Prezado(a) Senhor(a), Os alunos de mestrado da UNIRIO eventualmente realizam estudos experimentais para caracterizar/avaliar suas pesquisas. Este estudo é conduzido por Cesar Barbosa e William Brum sob a orientação do professor Gleison Santos. Você foi previamente selecionado pelo seu conhecimento/experiência e está sendo convidado a participar desta pesquisa.</p> <p>1. Procedimentos: O estudo será realizado com data e hora marcada com os participantes pré-selecionados. Para participar do estudo normalmente será aplicado um formulário de caracterização de perfil, a fim de identificar seu nível de conhecimento/experiência. Em seguida, o estudo será executado de forma individual ou em grupos formados, seguindo sempre o planejamento do estudo feito pelo pesquisador(a) responsável. Caso seja necessário, ao final do estudo será solicitado ao participante que responda um questionário de avaliação sobre a tecnologia de software que está sendo caracterizada/avaliada.</p> <p>2. Tratamento de possíveis riscos e desconfortos: Serão tomadas todas as providências durante a coleta de dados de forma a garantir a sua privacidade e seu anonimato.</p> <p>3. Benefícios e Custos: Espera-se que, como resultado deste estudo, você possa aumentar seus conhecimentos, de maneira a contribuir para o aumento da qualidade das atividades com as quais você trabalhe ou possa vir a trabalhar. Este estudo também contribuirá com resultados importantes para a pesquisa que está sendo realizada. Você não terá nenhum gasto ou ônus com a sua participação no estudo e também não receberá qualquer espécie de reembolso ou gratificação devido à autorização dos seus dados na pesquisa.</p> <p>4. Confidencialidade da Pesquisa: Toda informação coletada neste estudo é confidencial e seu nome não será identificado de modo algum, a não ser em caso de autorização explícita para este fim. Quando os dados forem coletados, seu nome será removido dos mesmos e não será utilizado em nenhum momento durante a análise ou apresentação dos resultados.</p> <p>5. Participação: Sua participação neste estudo é muito importante e voluntária, pois requer a sua aprovação para utilização dos dados coletados neste estudo. Você tem o direito de não querer participar ou de sair deste estudo a qualquer momento, sem penalidades. Em caso de você decidir se retirar do estudo, favor notificar o pesquisador(a) responsável. Os pesquisadores responsáveis pelo estudo poderão fornecer qualquer esclarecimento sobre o mesmo, assim como tirar dúvidas. william.brum@uniriotec.br / cesar.silva@uniriotec.br</p> <p>6. Declaração de Consentimento: Declaro que li e estou de acordo com as informações contidas neste documento e que toda linguagem técnica utilizada na descrição deste estudo de pesquisa foi explicada satisfatoriamente, recebendo respostas para todas as minhas dúvidas. Confirmando também que recebi uma cópia deste Termo (TCLE), compreendo que sou livre para não autorizar a utilização dos meus dados neste estudo em qualquer momento, sem qualquer penalidade. Declaro ter mais de 18 anos e concordo de espontânea vontade em participar deste estudo.</p>			
_____ Participante		_____ William Brum	
		_____ Cesar Barbosa	

Figure 40 - Termo de Consentimento para participação do estudo de caso

Além disso, a Figura 41 apresenta o questionário formulado para o estudo de caso. As questões de 1 a 3 são referentes ao tópico Facilidade de Uso do AMP.br; questões de 4 a 7 ao tópico Utilidade do AMP.br na elaboração de um plano de medição e avaliação dos resultados; por fim, as questões de 8 a 10 ao tópico Comparação entre os modelos apresentados pelo AMP.br e os disponíveis no Github.

		AVALIAÇÃO DE APLICABILIDADE DO TEMPLATE AMP.br Universidade Federal do Estado do Rio de Janeiro, Centro de Ciências Exatas e Tecnologia.	
Pesquisadores Cesar Barbosa e William Brum		Orientadores: Gleison Santos, D.Sc.	
Participante		Nome	
		Email	
Descrição do Estudo			
Questões			Resposta
Q1	Foi fácil identificar as influências geradas pelos indicadores nos objetivos da organização com o AMP.br?		
Q2	Foi fácil selecionar os indicadores adequados à necessidade do projeto consultando o conjunto de indicadores descrito com o AMP.br?		
Q3	Considera o AMP.br fácil de ser utilizado?		
Q4	Considera que o AMP.br melhoraria o desempenho na avaliação de quais indicadores devem ser selecionados para um plano de medição?		
Q5	Considera que o uso do AMP.br melhoraria o desempenho na análise dos resultados das medições?		
Q6	Considera o AMP.br útil para montar um plano de medição?		
Q7	Considera o AMP.br útil para a padronização do uso de indicadores na organização?		
Q8	Considera os modelos apresentados pelo AMP.br complementares às medições fornecidas pelo Github?		
Q9	Consideraria o AMP.br um substituto às medições fornecidas pelo Github?		
Q10	Considera o AMP.br mais conveniente para elaboração de planos de medição do que as medições do Github?		
Respostas Possíveis: 0 = "Não tenho elementos para responder" 1 = "Concordo Totalmente" 2 = "Concordo Parcialmente" 3 = "Discordo Parcialmente" 4 = "Discordo Totalmente"			

Figure 41 - Avaliação de aplicabilidade do estudo de caso

4.2.2 Identificação de um projeto ativo de desenvolvimento de software no Github

A organização escolhida para o estudo de caso foi um grupo de pós graduação da Universidade Federal Fluminense (UFF) que mantém vários repositórios no Github de diversos projetos de desenvolvimento. O projeto escolhido foi o noWorkflow devido ao número de *issues* criadas pelos desenvolvedores, utilização frequente de rótulos nas *issues* e alto nível de atividade no projeto.

O noWorkflow é um projeto que visa permitir pesquisadores se beneficiarem de análise de dados mesmo que não usam um sistema de fluxos de trabalho. Além disso, o objetivo é evitar usar convenções de nomenclatura para armazenar arquivos gerados em execuções prévias. O projeto foi desenvolvido em Python, com suporte às versões 2.7 e 3.5, com o intuito de permitir a coleta de dados sem utilizar um sistema de controle de versionamento ou outro ambiente.

Ademais, a equipe é composta por seis colaboradores do Brasil e dos Estados Unidos, contudo apenas um colaborador brasileiro conseguiu ser entrevistado para o estudo de caso, então não pode ser garantida a opinião sobre o AMP.br para toda a equipe envolvida. Todos os indicadores desenvolvidos no trabalho se adequam ao cenário do estudo de caso e foram avaliados pelo desenvolvedor entrevistado, junto dos objetivos estratégicos e de medição.

4.2.3 Análise do perfil profissional dos participantes

Conforme mencionado anteriormente na seção 4.2.1, foram identificados seis colaboradores a princípio no projeto, entretanto apenas um colaborador pode ser entrevistado. O perfil do entrevistado se encontra na Tabela 5:

Tabela 6 - Perfil do Participante 1

Participante	Perfil
Participante 1	Desenvolvedor

O Participante 1 é o principal desenvolvedor do projeto, portanto possui conhecimento sobre as necessidades do projeto em termos de indicadores e pode avaliar as informações proporcionadas pelo *template*. Possui alguns anos de experiência com desenvolvimento em Python.

Além disso, também foi entrevistado um gerente de projetos com experiência na análise e elaboração de indicadores. O perfil do entrevistado se encontra na tabela 6

Tabela 7 - Perfil do Participante 2

Participante	Perfil
Participante 2	Gerente de Projeto

O Participante 2 é um gerente de projetos com experiência em elaboração e análise de indicadores de projetos de software, portanto considerou-se que ele é adequado a avaliar tanto a aplicação de um *template* como o AMP.br como uma ferramenta de apoio ao Github quanto na relevância dos indicadores elaborados.

4.2.4 Procedimento de Entrevista com o Desenvolvedor

Após agendar uma reunião no dia 01/06/2016 na UFF, foi apresentado ao Participante 1 o *template* AMP.br já preenchido com os dados referentes ao projeto

NoWorkflow. Durante a reunião foi explicado o modelo, comentários foram registrados e possíveis melhorias foram apresentadas, e o questionário na Figura 42 foi preenchido e suas respostas foram consolidadas na subseção seguinte. A entrevista com o participante 2 foi marcada no dia 03/06/2016 e o questionário não foi aplicado por ele não ser parte da equipe do estudo de caso analisado e já possuir conhecimento sobre o AMP.br.

O processo da entrevista foi:

- A reunião começou com uma explicação sobre o estudo de caso, o que era o AMP.br e a razão dos indicadores gerados serem os apresentados ao entrevistado;
- Após a apresentação, o material disponível no Apêndice 2 foi apresentado ao entrevistado para avaliação;
- Durante a análise, o entrevistado tirou dúvidas e sugeriu melhorias;
- Após a análise, foram apresentados os indicadores existentes atualmente no Github;
- Com a análise dos indicadores do Github realizada, o questionário de aplicabilidade foi preenchido pelo entrevistado.

4.2.5 Análise dos resultados da Avaliação com o Desenvolvedor

Com o objetivo de analisar a estrutura e relevância, foi entrevistado um membro participante do projeto utilizado no GitHub com a função de desenvolvedor no projeto selecionado para o estudo de caso. Como já abordado, o *template* é dividido em três partes: Perfil Organizacional, Mapeamento de Indicadores e Fichas de Indicadores e seguiremos com as considerações externas nesta ordem.

A análise foi consolidada através do preenchimento da ficha denominada “Avaliação de aplicabilidade do template AMP.br” e abaixo temos o seu preenchimento.

Neste preenchimento notamos que o entrevistado notou certa dificuldade para identificar as influências geradas pelos indicadores nos objetivos da organização utilizando o template (Q1), da mesma forma que não considerou fácil selecionar os indicadores adequados à necessidade do projeto consultando o conjunto de indicadores descritos com o template (Q2).

Tabela 8 - Avaliação de Aplicabilidade preenchida pelo entrevistado

Questões		Resposta
Q1	Foi fácil identificar as influências geradas pelos indicadores nos objetivos da organização com o AMP.br?	2
Q2	Foi fácil selecionar os indicadores adequados à necessidade do projeto consultando o conjunto de indicadores descrito com o AMP.br?	3
Q3	Considera o AMP.br fácil de ser utilizado?	1
Q4	Considera que o AMP.br melhoraria o desempenho na avaliação quais indicadores devem ser selecionados para um plano de medição?	2
Q5	Considera que o uso do AMP.br melhoraria o desempenho na análise dos resultados das medições?	0
Q6	Considera o AMP.br útil para montar o plano de medição?	1
Q7	Considera o AMP.br útil para a padronização do uso de indicadores na organização?	1
Q8	Considera os modelos apresentados pelo AMP.br complementares às medições fornecidas pelo Github?	2
Q9	Consideraria o AMP.br um substituto às medições fornecidas pelo Github?	2
Q10	Considera o AMP.br mais conveniente para elaboração de planos de medição do que às medições do Github?	2
Respostas Possíveis: 0 = “Não posso Opinar” 1 = “Concordo Totalmente” 2 = “Concordo Parcialmente” 3 = “Discordo Parcialmente” 4 = “Discordo Totalmente”		

Porém, o entrevistado considerou fácil a utilização do template (Q3) e concordou parcialmente que a utilização do mesmo melhoraria o desempenho na avaliação de quais indicadores deveriam ser selecionados para um plano de medição (Q4).

O entrevistado não pôde tecer considerações sobre a questão sobre se o uso do template melhoraria o desempenho na análise dos resultados das medições devido a não possuir experiência com essa área do projeto (Q5), porém, considera o AMP.br útil para montar o plano de medição (Q6) e útil para a padronização do uso de indicadores na organização (Q7).

O entrevistado concorda parcialmente os modelos apresentados pelo AMP.br como complementares às medições fornecidas pelo Github (Q8), da mesma forma como concorda parcialmente que o template seria um substituto às medições fornecidas pelo Github (Q9). Além disso, o entrevistado concorda parcialmente que o AMP.br seria mais conveniente para elaboração de planos de medição do que às medições do Github (Q10).

Os componentes *Perfil Organizacional* e *Mapeamento dos Indicadores* e sua subdivisão: Mapa de Objetivos e Indicadores, Mapa para o MR-MPS-SW e Mapa de Indicadores, observamos que o entrevistado apreciou a forma com que estes são

desenhados e organizados pois estes possuem maior visibilidade que os indicadores do Github.

Dentro do componente **fichas de indicadores**, o entrevistado gostou da forma que o elemento **Indicador** da ficha é apresentado da mesma forma que o elemento **Necessidade de informação**. Em relação a categoria, ele nota uma dificuldade na compreensão entre as categorias e a relação entre o impacto e a influência destas.

O Participante 1 também gostou, seja na ficha quanto no mapa, da forma com que os **objetivos** são exibidos, pois garantem maior compreensão do gráfico. Isto é uma característica que aumenta a relevância deste em relação aos gráficos do próprio github.

O **nível MR-MPS-SW** foi apreciado também pelo entrevistado, o que é condizente com o intuito desta ficha, onde espera-se que a presença dos níveis e processos apoiados sejam relevantes para alinhar a medição.

A parte do elemento **Representação Gráfica** foi a mais analisada e posta em consideração visto ser a parte em que o entrevistado visualiza os dados da coleta graficamente. Ele considerou relevante esta apresentação, mas gostaria de uma possível análise histórica da abertura e fechamento de *issues*.

Os três procedimentos: **Procedimento de Medição e Análise**, **Procedimento de Análise Baseado em Critérios** e **Procedimento de Análise Baseado em Relacionamentos** foram considerados relevantes para com os trabalhos e são a chave para que estas análises sejam complementares às medições fornecidas pelo Github.

Ademais, o entrevistado sugeriu uma possível melhoria ao AMP.br: poderia existir mais um mapa ou tabela que separasse os indicadores por suas categorias e, dentro das categorias, estivessem ordenados considerando a influência do indicador em determinada categoria.

4.2.6 Análise da entrevista com o Gerente de projeto

O Participante 2 considerou os indicadores apresentados satisfatórios em relação a suprimir as necessidades de um gerente de projetos em um projeto *open source* no Github. Todavia, algumas sugestões de melhorias nos indicadores, e de indicadores adicionais, foram levantadas durante a entrevista.

Uma das sugestões foi de adicionar, em todos os projetos do Github, rótulos de “Fora do Escopo” e “Duplicado” a fim de garantir uma visão melhorada dos indicadores RIC, IAF e NIA. Com esses rótulos seria possível explicar a razão de algumas *issues* estarem sem responsável e fechadas. Outra melhoria seria incluir mais uma premissa no

indicador TIF na seção Procedimento de Análise baseado em critérios referente a um tempo máximo, definido pelos gerentes, para reavaliação das *issues* baseado no tempo de fechamento delas.

Por fim, a sugestão de melhoria do Participante 1 referente a tabela de ordenação de categorias foi levada ao Participante 2 e este concordou com a sugestão, indicando que poderia ser implementada numa futura revisão do AMP.br. Além disso, o Participante 2 confirmou a aplicação do AMP.br como válida dado o contexto dos indicadores e do Github, julgando factível usar o *template* e os indicadores elaborados como complementares aos já existentes no Github.

4.3 Considerações Finais

A avaliação mostrou bons resultados em utilizar os indicadores propostos, através do *template* AMP.br, para complementar os indicadores já existentes no Github. Realizar a entrevista com um dos desenvolvedores principais foi interessante por permitir uma visão diferenciada de um gerente de projetos sobre os indicadores necessários para um projeto no Github.

A melhoria proposta pelo Participante 1 é pertinente, pois permite o usuário do AMP.br filtrar os indicadores relacionados a uma categoria específica rapidamente, facilitando a análise de um conjunto de indicadores em determinada categoria.

A entrevista com um gerente de projeto também provou ser proveitosa, pois ele pôde afirmar que os indicadores elaborados atendem de fato as principais necessidades dos gestores de projeto, mesmo que precisem de algumas modificações, e que o AMP.br foi aplicado corretamente e tem utilização em padronizar os indicadores gerados para o Github.

Uma limitação da avaliação realizada foi não ter sido possível entrevistar o gerente de projetos do noWorkflow para conseguir outra visão sobre os indicadores e a aplicação do *template*. A sugestão do Participante 1 não foi implementada após a entrevista e foi considerada como um trabalho futuro. No capítulo seguinte serão abordadas as considerações finais, bem como limitações e trabalhos futuros.

5. Conclusão

Neste capítulo são resumidos o objetivo do trabalho, as atividades realizadas a fim de alcançar o objetivo e a análise do resultado obtido. Ademais, são apresentados possíveis trabalhos futuros e limitações encontradas na execução do trabalho.

5.1 Considerações Finais

O objetivo deste trabalho foi a definição de indicadores para apoiar a gestão de projetos *open source* no Github e padronizá-los através do AMP.br a fim de simplificar o seu entendimento para os envolvidos no projeto. Para atingir tal objetivo, foi desenvolvida uma aplicação em Python, chamada de PyNdicators, com o propósito de coletar informações dos repositórios do Github para elaborar os indicadores. Foram definidos objetivos estratégicos e de medição baseados nas informações coletadas e, com o uso do AMP.br, foram definidos indicadores adequados a este contexto.

O trabalho apresentou também avaliações do uso dos indicadores padronizados pelo AMP.br a fim de avaliar se o uso do *template* complementar ou não os indicadores e a forma como são apresentados no Github. Por meio da avaliação realizada, foi constatado que o *template* seria complementar às medições do Github e poderia, parcialmente, substituí-las com algumas modificações.

5.2 Limitações do Trabalho

O trabalho possuiu algumas limitações que devem ser levadas em considerações para sua avaliação e continuidade, sendo essas:

- O pacote *github3* ainda está em desenvolvimento e a API do Github é limitada em alguns aspectos sobre as informações que podem ser extraídas, como o número de downloads realizados de um arquivo específico;
- Só foi possível entrevistar um colaborador do projeto alvo do estudo de caso;
- Devido às informações que foram extraídas, só foi possível elaborar um número limitado de indicadores que, apesar de englobar uma parcela significativa das necessidades de um gerente de projeto, não atende a todas as visões necessárias.
- Existe a dependência da planilha eletrônica para atualizar os indicadores gerados a partir das bases extraídas. O ideal seria um sistema que lê automaticamente essas bases a partir de um banco e atualiza os indicadores em tempo real.

5.3 Trabalhos Futuros

Durante a execução do trabalho, foi possível identificar oportunidades de trabalhos futuros, tais como:

- Criação de uma visão para separar os indicadores por categoria e ordená-los por influência / impacto, conforme sugerido no estudo de caso realizado;
- Desenvolver indicadores de dados históricos acumulados ao longo dos meses;
- Prever indicadores que possam ser utilizados se o Github permitir a criação de campos customizados de *issues*, como complexidade / pontos associados;
- Aplicar a padronização de indicadores e os indicadores elaborados em outros repositórios de gits, como o Bitbucket;
- Desenvolvimento de métodos para o pacote *github3* a fim de possibilitar o acesso a algumas informações que não conseguiram ser resgatadas, como número de linhas alteradas por arquivo;
- Desenvolvimento de uma ferramenta que permita automatizar o processo de geração da planilha eletrônica de indicadores.

6. Referências Bibliográficas

BARCELLOS, M. P. Uma Estratégia para Medição de Software e Avaliação de Bases de Medidas para Controle Estatístico de Processos de Software em Organizações de Alta Maturidade. 2009. Tese (Doutorado em engenharia de sistemas e computação). UFRJ, Rio de Janeiro, 2009.

BARCELLOS, M. P., FALBO, R. A. AND ROCHA, A. R., A Strategy for Preparing Software Organizations for Statistical Process Control. Journal of the Brazilian Computer Society, 2013.

BASILI, V., HEIDRICH, J., LINDVALL, M., MÜNCH, J., REGARDIE, M., ROMBACH, D., et al.: GQM+Strategies: A comprehensive methodology for aligning business strategies with soft-ware measurement, Kaiserslautern, Germany, MetriKon 2007 (2007) 1–2

BASILI, V. R., ROMBACH, H. D., Measurement, Encyclopedia of Software Engineering, John Wiley & Sons, 1994.

BASS, L. Constructing Superior Software; Applying Proven Practices. (P. Clements, Ed.). Thousand Oaks, CA, USA: New Riders Publishing. 1999.

BPMN, 2016, Disponível em: <http://www.bpmn.org> Acesso em: 06 abr. 2016.

BONELLI , S. Uma abordagem para registro de indicadores de medição em Organizações de desenvolvimento de software, Dissertação de mestrado 2014

CANFORA, G., GARCIA, F., PIATTINI, M., RUIZ, F., VISAGGIO, C. A., 2004, "A Family of Experiments to Validate Metrics for Software Process Models", *The Journal of Systems and Software*, v. 77, n. 2, pp. 113-129.

CMMI Product Team. CMMI for Development, Version 1.3 (CMU/SEI-2010-TR-033). Software Engineering Institute, Carnegie Mellon University, 2010. <http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=9661>. Acesso em: 02/06/2016.

DUMKE, R. R., BRAUNGARTEN, R., BLAZEY, M., HEGEWALD, H., REITZ, D., DRICHTER, K., 2006, *Software Process Measurement and Control - A Measurement-based Point of View of Software Processes*, Technical Report, Dept. of Computer Science, University of Magdeburg, Germany.

FERNANDES, A. A., "Gerência de Software Através de Métricas". Atlas, São Paulo, São Paulo, 1995.

FENTON, N. E., PFLEEGER, S. L., 1997, *Software Metrics: A Rigorous and Practical Approach*, PWS Publishing Company.

ISO/IEC. ISO/IEC 15504-2. (E) Information technology – Process assessment, International Organization for Standardization and the International Electrotechnical Commission, Geneva, Switzerland. 2002.

ISO/IEC. ISO/IEC 15939. (E) Software Engineering – Software Measurement Process, International Organization for Standardization and the International Electrotechnical Commission, Geneva, Switzerland. 2002.

KAN, S. *Metrics and models in software engineering*. 2.ed. Boston: Addison Wesley, 2003.

MCGARRY, J.; et al. *Practical Software Measurement: objective information for decision makers*. 1. ed. Boston: Addison-Wesley, 2002.

MCGARRY, J., CARD, D., JONES C., LAYMAN, B., CLARK, E., DEAN, J., HALL, F. Practical Software Measurement: Objective Information for Decision Makers. 2008.

MONTEIRO, L. F. S.; OLIVEIRA, K. M. DE. Defining a catalog of indicators to support process performance analysis. Journal of Software Maintenance and Evolution: Research and Practice, v. 23, n. 6, p. 395–422, 2011. John Wiley & Sons, Ltd. Disponível em: <<http://dx.doi.org/10.1002/smr.482>>. Acesso em: 08 abr.2016.

PMI, Project Manager Institute. PMBOK - A Guide to the Project Management Body of Knowledge. 5a. ed. Newtown Square, Pensilvânia: Project Management Institute, Inc., 2013.

ROCHA, A. R. C., SOUZA, G. S., BARCELLOS, M. P. Medição de Software e Controle Estatístico de Processos. 2012.SANTOS, B. O. Utilização de métricas nos projetos de desenvolvimento de sistemas de informação: um survey com gerentes de projetos, 2009. UNIRIO - Universidade Federal Do Estado Do Rio De Janeiro. Disponível em: <http://www2.uniriotec.br/ppgi/banco-de-dissertacoes-ppgi-unirio/ano-2009/utizacao-de-metricas-nos-projetos-de-desenvolvimento-de-sistemas-de-informacao-um-survey-com-gerentes-de-projetos/at_download/file> Acesso em: 05 abr.2016.

SEI – Software Engineering Institute. Applications of the Indicator Template for Measurement and Analysis. Pittsburgh, PA, EUA 2004. Disponível em: <<http://www.sei.cmu.edu/pub/documents/04.reports/pdf/04tn024.pdf>>. Acesso em: 05 abr.2016.

TAKARA, A.; BETTIN, A.; TOLEDO, C. Problems and Pitfalls in a CMMI level 3 to level 4 Migration Process, In: Sixth International Conference on the Quality of Information and Communications Technology, 2007, Lisboa, Portugal, Anais... Disponível em: <http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?isnumber=4335220&arnumber=4335237&count=33&index=16>. Acesso em: 05 abr. 2016.

SOFTEX, 2016, MPS.BR: Melhoria de Processo do Software Brasileiro - Guia Geral : 2016 Disponível em: <http://www.softex.br/mpsbr>. // http://www.softex.br/wp-content/uploads/2016/04/MPS.BR_Guia_Geral_Software_2016-com-ISBN.pdf Acesso em: 07 mai. 2016.

WEBER, C., LAYMAN, B., 2002, "Measurement Maturity and the CMMI: How Measurement Practices Evolve as Process Mature", American Society for Quality Magazines & Journals, v. 4, n.3, pp. 7-20.

7. Apêndice 1 – Aplicação PyNdicators

```

# coding=utf-8
import csv
import github3

git_username = 'wbrum'
git_api_token = '*****'

# csv name
csv_name = "git_hub_stats.csv"

def run_csv():
    # Exporta estatísticas de um repositório em formato CSV
    output_csv = csv.writer(open(csv_name, 'wb'), delimiter=',')
    github = github3.login(username=git_username, token=git_api_token)

    # csv headers
    headers = [
        'author',
        'total',
        'Start of the Week',
        'Deletions',
        'Additions',
        'Commits',
    ]

    # escreve header rows
    output_csv.writerow(headers)

    # baixa estatísticas do repositório e transcreve para csv
    git_stats = github.repository("gems-uff",
"noworkflow").iter_contributor_statistics()
    for git_stat in git_stats:
        for git_week in git_stat.alt_weeks:
            commit = [
                git_stat.author,
                git_stat.total,
                git_week['start of week'],
                git_week['deletions'],
                git_week['additions'],
                git_week['commits'],
            ]
            output_csv.writerow(commit)

if __name__ == '__main__':
    run_csv()

```

Módulo 1 - Modulo de extração de Estatísticas da API do Github

```

# coding=utf-8
import csv
import github3

git_username = 'wbrum'
git_api_token = '*****'

# csv name
csv_name = "git_hub_commits.csv"

def run_csv():
    # Exporta commits de um repositório em formato CSV

    output_csv = csv.writer(open(csv_name, 'wb'), delimiter=',')
    github = github3.login(username=git_username, token=git_api_token)
    # csv headers
    headers = [
        'author',
        'message',
        'date',

    ]

    # escreve header rows
    output_csv.writerow(headers)

    # baixa commits e transcreve para csv
    git_commits = github.repository("gems-uff", "noworkflow").iter_commits()
    for git_commit in git_commits:
        print git_commit.commit.message
        commit = [

            git_commit.author,
            git_commit.commit.message.encode('utf8'),
            git_commit.commit.author['date'],
        ]
        output_csv.writerow(commit)

if __name__ == '__main__':
    run_csv()

```

Módulo 2 - Módulo de Extração de *Commits* da API do Github

```

# coding=utf-8
import csv
import github3

git_username = 'wbrum'
git_api_token = '*****'
state = 'all' # 'open','closed' ou 'all'

# csv name
csv_name = "git_hub_issues.csv"

def run_csv():
    # Exporta issues de um repositório em formato CSV
    output_csv = csv.writer(open(csv_name, 'wb'), delimiter=',')
    github = github3.login(username=git_username, token=git_api_token)

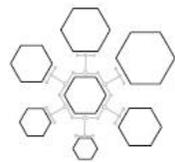
    # csv headers
    headers = [
        'id',
        'title',
        'state',
        'creator',
        'labels',
        'created_at',
        'updated_at',
        'closed_at',
        'assignee',
    ]
    # escreve header rows
    output_csv.writerow(headers)
    # pega issues e transcreve para csv
    git_issues = github.iter_repo_issues('gems-uff', 'noworkflow', "", state)
    for git_issue in git_issues:
        print git_issue.title
        issue = [
            git_issue.number,
            git_issue.title.encode('utf8'),
            git_issue.state,
            git_issue.user,
            git_issue.labels,
            git_issue.created_at,
            git_issue.updated_at,
            git_issue.closed_at,
            git_issue.assignee,
        ]
        output_csv.writerow(issue)

if __name__ == '__main__':
    run_csv()

```

Módulo 3 - Módulo de Extração de *Issues* da API do Github

8. Apêndice 2 – Conjunto de Indicadores Criados



AMP [br]

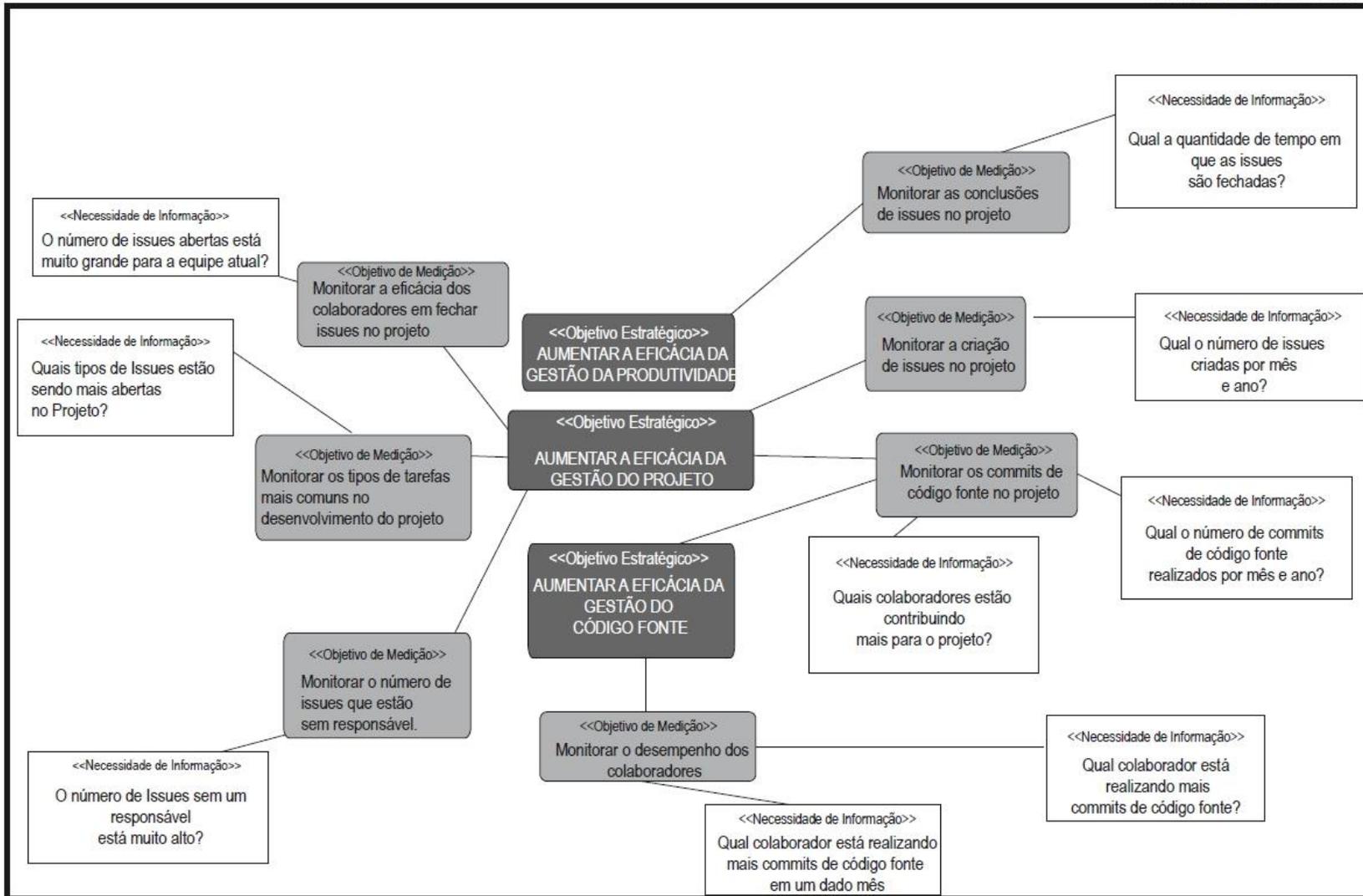
Apoio à Medição de Projetos
de Software

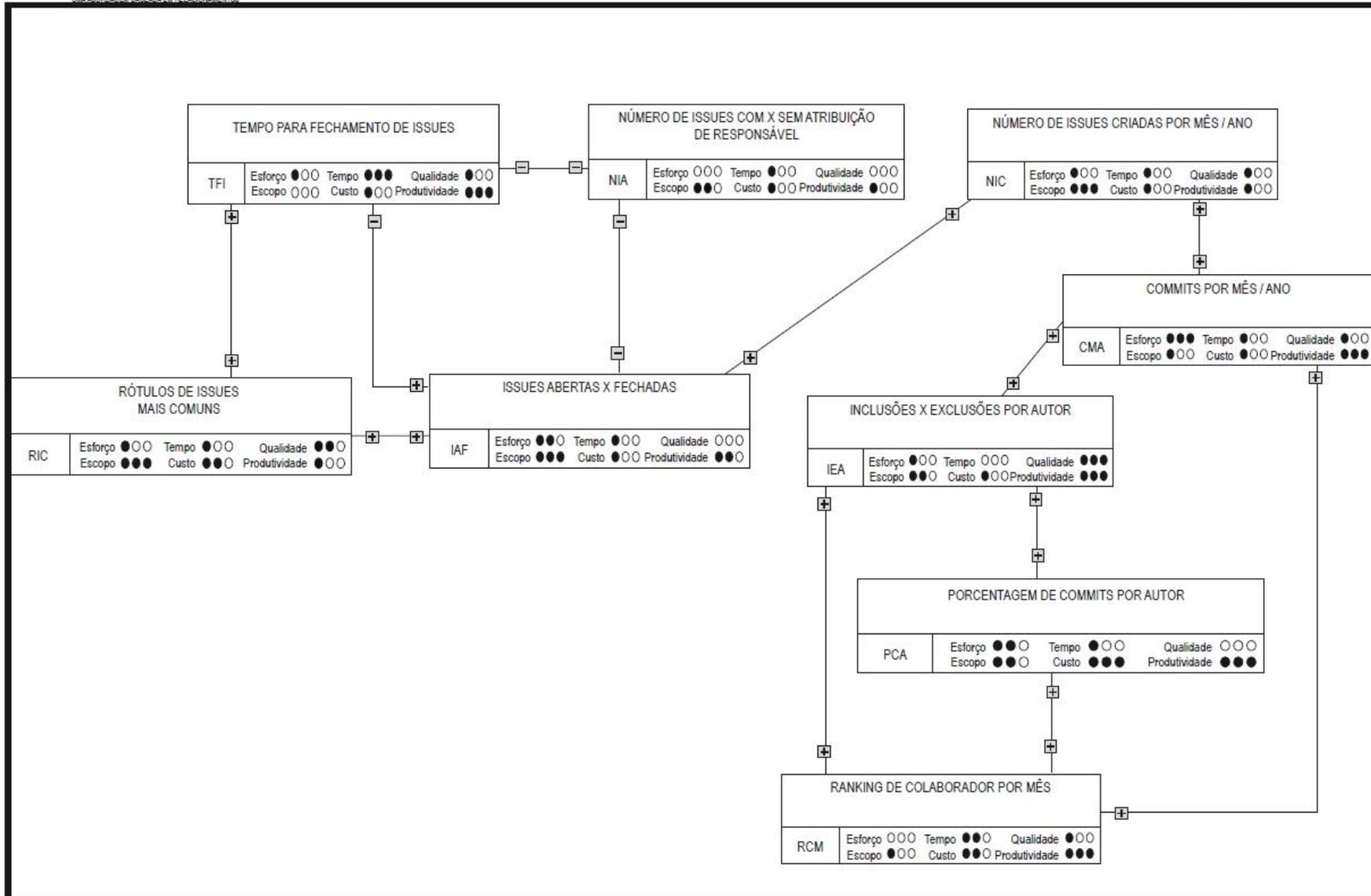
UMA ABORDAGEM BASEADA EM RELACIONAMENTOS

UFF

Instituto de Pós Graduação em Ciência da Computação

Repositório de Medição: <https://github.com/gems-uff/noworkflow/>





INDICADORES DE PATTERN		MR-MPS-SW	
		G	F
SIGLA	NOME	GPR	GCO
RIC	RÓTULO DE ISSUES MAIS COMUNS	X	
IAF	ISSUES ABERTAS X FECHADAS	X	
NIA	NÚMERO DE ISSUES COM X SEM RESPONSÁVEL	X	
TFI	TEMPO PARA FECHAMENTO DE ISSUES	X	
NIC	NÚMERO DE ISSUES CRIADOS POR MÊS / ANO	X	
CMA	COMMITTS POR MÊS / ANO		X
IEA	INCLUSÕES X EXCLUSÕES POR AUTOR		X
PCA	PORCENTAGEM DE COMMITTS POR AUTOR		X
RCM	RANKING DE COLABORADOR POR MÊS		X

FICHA DE INDICADORES
REPOSITÓRIO DE INDICADORES



AMP[br]

Apoio à Medição de Projetos de Software
UMA ABRIGADAGEM BASEADA EM RELACIONAMENTOS

Indicador

RIC – Rótulos de Issues Mais Comuns

Acompanha o andamento do projeto sob o aspecto de tipos de Issues. Permite ao gerente visualizar qual tipo de tarefa está sendo trabalhada com maior frequência

Necessidade de Informação

Quais tipos de Issues estão sendo mais abertas no Projeto?

Categorias

Esforço: ● ○ ○
 Escopo: ● ● ● ○
 Tempo: ● ○ ○ ○

Custo: ● ● ● ○
 Qualidade: ● ● ● ○
 Produtividade: ● ● ● ○

Objetivo Estratégico

Objetivos: Aumentar a eficácia da gestão do projeto

Objetivo de Medição

Monitorar os tipos de tarefas mais comuns no desenvolvimento do projeto

Tipo

controle
 qualidade
 desempenho

MR-MPS-SW
 NÍVEL: G
 PROC: GPR

Representação Gráfica do Indicador

Contagem de labels

Rótulos de Issues mais Comuns

Label	closed	open
[]	3	3
[bug]	29	5
[enhancement, related work]	1	1
[enhancement]	32	27
[Library, related work]	2	2
[Library]	3	3
[related work]	0	8

state: closed (blue), open (red)

labels: []

Procedimento de Medição e Análise

MEDIÇÃO

Status atualizado do Projeto

INÍCIO: Em qualquer fase do projeto, respeitando a frequência de execução do indicador

Medida: RIC

Coletar dados sobre os rótulos e estados das Issues na área de Issues do Github através do PyNdicators

Atualizar dados na Planilha de Medição do Indicador

Planilha de Medição Atualizada

ANÁLISE

FIM

Executar o Procedimento de Análise Baseado em Relacionamentos

Avaliar gráfico gerado e executar o Procedimento de Análise Baseado em Critérios disponibilizado na Ficha de Descrição do Indicador

Geração automática do gráfico para a visualização das Labels de Issues mais comuns na planilha de medição (RIC)

Selecionar dados relevantes para a análise. (Especificar período de medição e/ou tipos/estados de issues).

Procedimento de Análise Baseado em Relacionamentos

Indicador: RIC

IAF: Issues Abertas x Fechadas (+)

TFI: Tempo Médio de Fechamento de Issues (+)

O aumento no indicador de Issues Abertas x Fechadas impacta diretamente em diversos tipos de issues aumentarem devido ao número crescente de issues criadas. Desta forma, o gerente de projeto deve avaliar qual o tipo de issue que está sendo mais criado.

O gerente do projeto pode avaliar quais tipos de issues estão tomando mais ou menos tempo para serem concluídos e verificar com a equipe as possíveis complexidades desse tipo de issue. Um número crescente issues de um tipo específico também aumenta o número de issues que são eventualmente fechadas e o seu tempo de fechamento.

Medida Base: RIC: RÓTULOS ATRIBUÍDOS A CADA ISSUE

Periodicidade: MENSAL

Responsáveis: Coleta: GERENTE DO PROJETO; Análise: GERENTE DO PROJETO

ID	Premissa	Possível Conclusão	Possíveis Ações
1	Número de Issues Abertas é maior que Issues Fechadas em 10 unidades	Isso pode ocorrer devido a algum problema na execução do procedimento de cadastro de Issues, falta de experiência, número baixo de colaboradores na equipe, etc.	A equipe deve identificar quais atividades estão criando essas demandas de aberturas de issues e como resolver de maneira mais eficiente os problemas encontrados
2	Elementos no eixo X sem nome	Isso pode ocorrer devido a algum problema na execução procedimento de cadastro de Issues ou falta de experiência pelos criadores de issues	A equipe deve se reunir e identificar que issues estão sem tipo e definir em conjunto o tipo mais apropriado para cada uma.
3	Um tipo de issue apresenta 50% ou mais do total de tipos de issue	Um tipo de issue recorrente pode indicar um ciclo de melhorias na aplicação, devido a requisições constantes de novas funcionalidades, bem como pode indicar a presença de erros na aplicação	Ao identificar o tipo de Issue recorrente, a equipe deve-se reunir e compreender a razão daquele tipo aparecer muitas vezes. Se for do tipo bug, por exemplo, a equipe deve tomar mais atenção com a fase de testes durante o desenvolvimento
4	Número de Issues Fechadas é maior que Issues Abertas em 10 unidades	Os colaboradores do projeto estão concluindo as Issues corretamente e resolvendo os possíveis problemas apresentados.	Não há ação necessária



Objetivos

Objetivo Estratégico: Aumentar a eficácia da gestão do projeto

Objetivo de Medição: Monitorar a eficácia dos colaboradores em fechar issues no projeto.

Tipo: controle, qualidade, desempenho

MR-MPS-SW

NÍVEL: **G**

PROC: GPR

Representação Gráfica do Indicador

mes_criacao: Contagem de state

Número de Issues Abertas x Issues Fechadas

Legend: [related work], [Library], [Library, related work], [enhancement], [enhancement, related work], [bug], []

Procedimento de Medição e Análise

MEDIÇÃO

INÍCIO: Em qualquer fase do projeto, respeitando a frequência de execução do indicador

Medida: IAF

Coletar dados sobre os estados das Issues na área de Issues do Github através do PyNdicators.

Atualizar dados na Planilha de Medição do Indicador

Planilha de Medição Atualizada

ANÁLISE

FIM: Executar o Procedimento de Análise Baseado em Relacionamentos

Avaliar gráfico gerado e executar o Procedimento de Análise Baseado em Critérios disponibilizado na Ficha de Descrição do Indicador

Geração automática do gráfico para a visualização dos estados das Issues na planilha de medição (IAF)

Selecionar dados relevantes para a análise. (Especificar o rótulo e/ou estados de issues e/ou Período de Medição).

Medida Base: IAF: STATUS ATRIBUÍDOS A CADA ISSUE | Períodicidade: MENSAL | Responsáveis: Coleta: GERENTE DO PROJETO | Análise: GERENTE DO PROJETO

Procedimento de Análise Baseado em Relacionamentos

Indicador IAF

RIC: Rótulos de Issues mais comuns

NIA: Número de Issues Com x Sem Responsável

TFI: Tempo para fechamento de Issues

NIC: Número de Issues criadas por Mês/Ano

O aumento no indicador de Issues Abertas x Fechadas impacta diretamente em diversos tipos de issues aumentarem devido ao número crescente de issues criadas. Desta forma, o gerente de projeto deve avaliar qual o tipo de issue que está sendo mais criado.

Um aumento no número de issues sem responsável pode impactar no número de issues abertas, pois os colaboradores podem não saber quem deve realizar a issue e ela não é fechada até alguém de fato resolvê-la

O gerente do projeto pode avaliar se o tempo médio de fechamento de issues está dentro do esperado a fim de não ter um backlog de Issues abertas fora do planejado

Meses com maior número de criação de issues aumentam o número total de issues abertas até as novas issues serem fechadas.

Procedimento de Análise Baseado em Critérios

ID	Premissa	Possível Conclusão	Possíveis Ações
1	Número de Issues Abertas é maior que Issues Fechadas em 10 unidades	Isso pode ocorrer devido a algum problema na execução de um determinado processo, falta de experiência, número baixo de colaboradores na equipe, etc.	A equipe deve identificar quem são os responsáveis pela resolução das issues, identificar os problemas e solucioná-los
2	Um tipo de issue apresenta 50% ou mais do total de tipos de issue	Um tipo de issue recorrente pode indicar um ciclo de melhorias na aplicação, devido a requisições constantes de novas funcionalidades, bem como pode indicar a presença de erros na aplicação	Ao identificar o tipo de Issue recorrente, a equipe deve-se reunir e compreender a razão daquele tipo aparecer muitas vezes. Se for do tipo bug, a equipe deve tomar mais atenção com a fase de testes durante o desenvolvimento
3	Issues sem tipo no gráfico	Isso pode ocorrer devido a algum problema na execução de um determinado processo ou falta de experiência pelos criadores de issues	A equipe deve se reunir e identificar que issues estão sem tipo e definir em conjunto o tipo mais apropriado para cada uma.
4	Número de Issues Fechadas é maior que Issues Abertas em 10 unidades	Os colaboradores do projeto estão concluindo as Issues corretamente e resolvendo os possíveis problemas apresentados.	Não há ação necessária



Objetivos

Objetivo Estratégico: Aumentar a eficácia da gestão do projeto

Objetivo de Medição: Monitorar o número de issues que estão sem responsável.

Tipo

controle qualidade
 desempenho

MR-MPS-SW NÍVEL: G
PROC.: GPR

Representação Gráfica do Indicador

Estados de issues

Número de Issues com Responsável x Sem Responsável

	closed	open
Sem responsável	50	39
Com responsável	14	10

Estados das issues

Procedimento de Medição e Análise

MEDIÇÃO

INÍCIO: Em qualquer fase do projeto, respeitando a frequência de execução do indicador

Medida: NIA

Coletar dados sobre os as issues e seus responsável e os estados das Issues na área de Issues do Github através do PyNdicators.

Atualizar dados na Planilha de Medição do Indicador

Planilha de Medição Atualizada

ANÁLISE

FIM: Executar o Procedimento de Análise Baseado em Relacionamentos

Avaliar gráfico gerado e executar o Procedimento de Análise Baseado em Critérios disponibilizado na Ficha de Descrição do Indicador

Geração automática do gráfico para a visualização das Issues com ou sem responsável e na planilha de medição. (NIA)

Selecionar dados relevantes para a análise. (Especificar com ou sem responsável e/ou estados de issues).

Medida Base: NIA: RESPONSÁVEIS ATRIBUÍDOS A CADA ISSUE | Periodicidade: MENSAL | Responsáveis: Coleta: GERENTE DO PROJETO | Análise: GERENTE DO PROJETO

Procedimento de Análise Baseado em Relacionamentos

Indicador NIA

IAF: Issues Abertas x Fechadas

Um aumento no número de issues sem responsável pode impactar no número de issues abertas, pois os colaboradores podem não saber quem deve realizar a issue e ela não é fechada até alguém de fato resolvê-la.

TFI: Tempo para fechamento de Issues

Issues que não possuem um responsável definido no momento da criação podem levar mais tempo para serem fechadas devido à equipe não ter conhecimento daquela tarefa ou não saber dar a devida priorização. Logo, uma diminuição no número de issues sem responsável pode também diminuir o tempo de fechamento das issues

Procedimento de Análise Baseado em Critérios

ID	Premissa	Possível Conclusão	Possíveis Ações
1	Número de Issues Fechadas sem Responsável está com 10 unidades a mais que Issues Fechadas com Responsável.	Isso pode ocorrer devido a algum problema na execução do procedimento de cadastro de Issues, como a habilidade com a ferramenta ou atenção na criação e/ou atribuição da issue.	A equipe deve ser orientada a detalhar a atribuição da issue aos colaboradores para controle dos mesmos.
2	Número de Issues Abertas Sem Responsável está com 10 unidades a mais que Issues Abertas com Responsável.	Isso pode ocorrer devido a algum problema na execução de um determinado processo, falta de experiência, número baixo de colaboradores na equipe, etc.	A equipe deve identificar qual a atividade pertinente a issue e atribuir um responsável.
3	Número de Issues Abertas com Responsável está com 10 unidades a mais que Issues Abertas sem Responsável	A equipe ou o gerente estão realizando a tarefa de cadastramento de responsável por issues de maneira adequada	Não há ação necessária
4	Número de Issues Fechadas com Responsável está com 10 unidades a mais que Issues Fechadas sem Responsável.		

PÁG.: x/xx

78



AMP[br]

Apoio à Medição de Projetos de Software
UMA ABORDAGEM BASEADA EM RELACIONAMENTOS

Indicador

TFI - Tempo para fechamento de issues

Acompanha o andamento do projeto sob o aspecto de tempo de fechamento de Issues. Permite o gerente visualizar quanto tempo as issues, no geral, estão levando para serem fechadas

Necessidade de Informação

Em quanto tempo em que as issues são fechadas?

Categorias

Esforço Custo
Escopo Qualidade
Tempo Produtividade

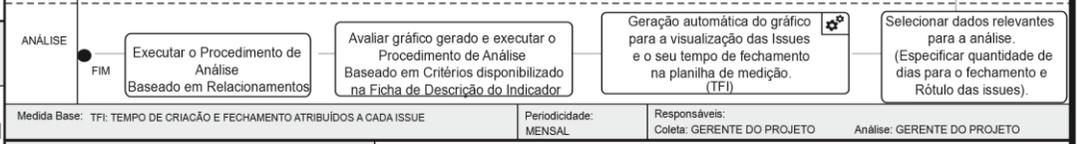
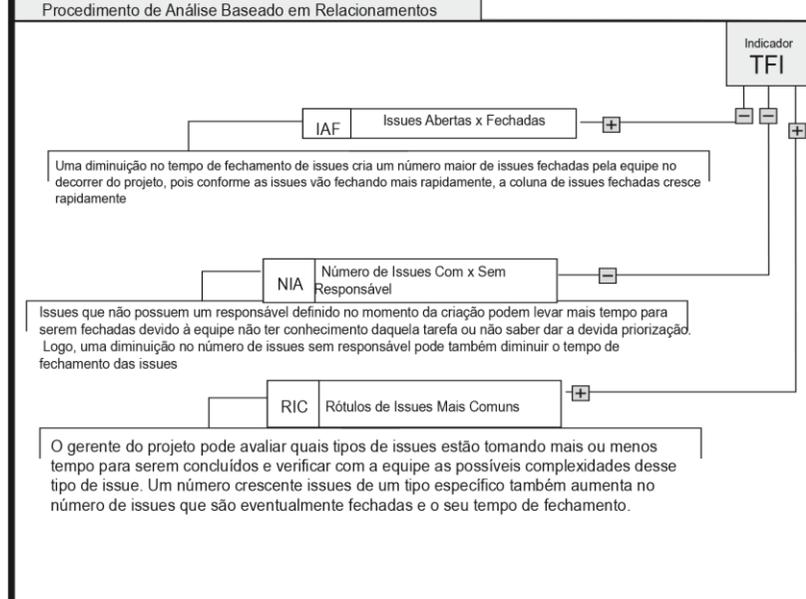
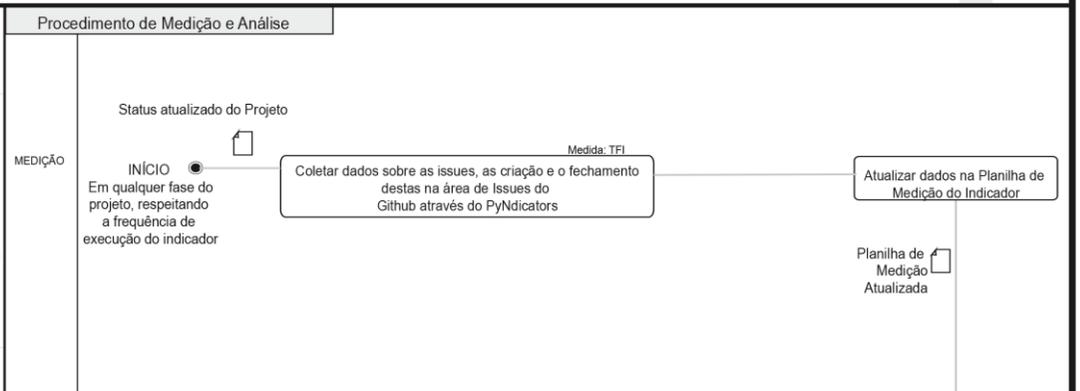
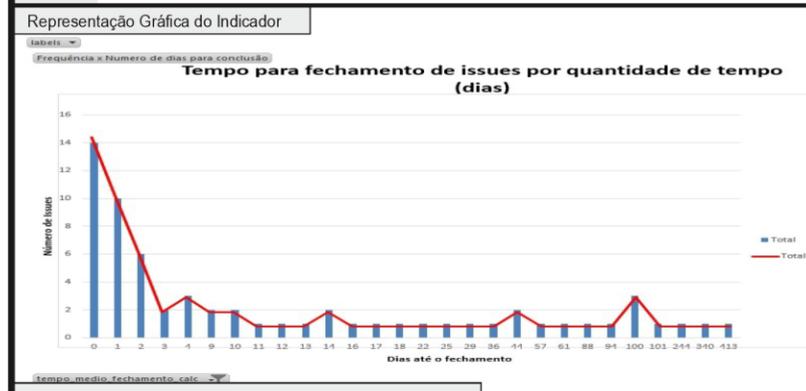
Objetivos

Objetivo Estratégico: Aumentar a eficácia da gestão da produtividade.

Objetivo de Medição: Monitorar as conclusões de issues no projeto.

Tipo: controle, qualidade, desempenho

MR-MPS-SW NÍVEIS: G PROC: GPR



Procedimento de Análise Baseado em Critérios

ID	Premissa	Possível Conclusão	Possíveis Ações
1	Existem colunas na direita do valor 60 do eixo X	A quantidade de dias que tem se levado para fechar as issues desde a sua criação está acima do limite. Verificando a tendência direcional do gráfico, podemos relacionar produtividade e foco em fases do projeto.	Podemos relacionar produtividade e foco em fases do projeto. Deve-se tomar ações de incentivo e direcionamento em caso de tendência para a região direita do gráfico e tentar entender quais tipos de issues estão causando essa condição
2	Existem colunas na esquerda do valor 60 do eixo X	A quantidade de dias que tem se levado para fechar as issues desde a sua criação está a baixo do limite. Verificando a tendência direcional do gráfico, podemos relacionar produtividade e foco em fases do projeto.	Não há ação necessária
3	Número de issues com tempo 0 de fechamento corresponde a 30% do total de issues fechadas	Colaboradores podem já possuir a solução para as issues e estão criando-as apenas para documentar a tarefa. Outra possibilidade é serem tarefas pequenas que estão sendo concluídas no mesmo dia pela equipe ou foram criadas por engano	Caso as tarefas estejam sendo criadas por engano, a equipe e o gestor precisam rever a forma e motivo de criar tarefas no projeto



AMP[br]

Apoio à Medição de Projetos de Software
UMA ABORDAGEM BASEADA EM RELACIONAMENTOS

Indicador

NIC - Número de Issues criadas por mês / ano

Acompanha o andamento do projeto sob o aspecto de abertura de tipos de issues durante o decorrer dos meses.

Necessidade de Informação

Qual o número de issues criados por mês e ano?

Categorias

Esforo ●●○○ Custo ●●○
Escopo ●●●○ Qualidade ●○
Tempo ●●○○ Produtividade ●○

<p>Objetivos</p> <p>Objetivo Estratégico Aumentar a eficácia da gestão da produtividade.</p> <p>Objetivo de Medição Monitorar as criações de issues no projeto.</p>	<p>Tipo</p> <p><input checked="" type="checkbox"/> controle <input type="checkbox"/> qualidade <input type="checkbox"/> desempenho</p> <p>MR-MPS-SW NÍVEL: G PROC: GPR</p>												
<p>Representação Gráfica do Indicador</p> <p>Contagem de mes_criacao</p> <p>Número de issues criadas por mês / ano</p> <p>mes_criacao</p>	<p>Procedimento de Medição e Análise</p> <p>MEDIÇÃO</p> <p>Status atualizado do Projeto</p> <p>INÍCIO ●</p> <p>Em qualquer fase do projeto, respeitando a frequência de execução do indicador</p> <p>Medida: NIC</p> <p>Coletar dados sobre as issues e a sua criação na área de Issues do Github através do PyNdicators.</p> <p>Atualizar dados na Planilha de Medição do Indicador</p> <p>Planilha de Medição Atualizada</p> <hr/> <p>ANÁLISE</p> <p>FIM ●</p> <p>Executar o Procedimento de Análise Baseado em Relacionamentos</p> <p>Avaliar gráfico gerado e executar o Procedimento de Análise Baseado em Critérios disponibilizado na Ficha de Descrição do Indicador</p> <p>Geração automática do gráfico para a visualização das Issues e o seu período de criação e as seus respectivos Rótulos na planilha de medição. (NIC)</p> <p>Selecionar dados relevantes para a análise (Especificar mês de criação das issues e os rótulos das issues).</p>												
<p>Procedimento de Análise Baseado em Relacionamentos</p> <p>Indicador NIC</p> <p>IAF Issues Abertas x Fechadas</p> <p>O número de Issues Abertas e Fechadas, quando relacionado à criação de issues permite o gestor do projeto analisar o aumento ou não dos backlogs, onde um número maior de aberturas e menor de fechamento fará o backlog crescer. Um backlog muito grande pode causar atraso na entrega de funcionalidades, além de tornar necessário a realização de uma priorização das atividades em espera para determinar as mais críticas.</p> <p>CMA Commits por mês x ano</p> <p>Os meses com maior número de issues podem resultar um aumento no número de commits de código fonte a fim de solucionar as tarefas atribuídas aos colaboradores</p>	<p>Medida Base: NIC: DATA DE CRIAÇÃO ATRIBUÍDA A CADA ISSUE</p> <p>Periodicidade: MENSAL</p> <p>Responsáveis: Coleta: GERENTE DO PROJETO Análise: GERENTE DO PROJETO</p> <p>Procedimento de Análise Baseado em Critérios</p> <table border="1"> <thead> <tr> <th>ID</th> <th>Premissa</th> <th>Possível Conclusão</th> <th>Possíveis Ações</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Um mês no gráfico possui um número de issues criadas acima do previsto pela equipe</td> <td>Meses maiores no gráfico são meses que há maior participação dos colaboradores no desenvolvimento de suas tarefas, como, por exemplo, projetos que utilizam issues para rastreamento de todas as tarefas do projeto</td> <td>Deve-se tomar ações de incentivo e direcionamento em caso de mês com baixa abertura de issues ou analisar a quantidade de bugs solucionados. Além disso, deve-se investigar os tipos das issues criadas nesses meses e entender com a equipe a razão da criação dessas issues</td> </tr> <tr> <td>2</td> <td>Um tipo de issue consiste de 50% ou mais das issues de um mês</td> <td>Um tipo de issue recorrente pode indicar um ciclo de melhorias na aplicação, devido a requisições constantes de novas funcionalidades, bem como pode indicar a presença de erros na aplicação</td> <td>Ao identificar o tipo de issue recorrente, a equipe deve-se reunir e compreender a razão daquele tipo aparecer muitas vezes. Se for do tipo bug, por exemplo, a equipe deve tomar mais atenção com a fase de testes durante o desenvolvimento a fim de diminuir o número de issues do tipo bug.</td> </tr> </tbody> </table>	ID	Premissa	Possível Conclusão	Possíveis Ações	1	Um mês no gráfico possui um número de issues criadas acima do previsto pela equipe	Meses maiores no gráfico são meses que há maior participação dos colaboradores no desenvolvimento de suas tarefas, como, por exemplo, projetos que utilizam issues para rastreamento de todas as tarefas do projeto	Deve-se tomar ações de incentivo e direcionamento em caso de mês com baixa abertura de issues ou analisar a quantidade de bugs solucionados. Além disso, deve-se investigar os tipos das issues criadas nesses meses e entender com a equipe a razão da criação dessas issues	2	Um tipo de issue consiste de 50% ou mais das issues de um mês	Um tipo de issue recorrente pode indicar um ciclo de melhorias na aplicação, devido a requisições constantes de novas funcionalidades, bem como pode indicar a presença de erros na aplicação	Ao identificar o tipo de issue recorrente, a equipe deve-se reunir e compreender a razão daquele tipo aparecer muitas vezes. Se for do tipo bug, por exemplo, a equipe deve tomar mais atenção com a fase de testes durante o desenvolvimento a fim de diminuir o número de issues do tipo bug.
ID	Premissa	Possível Conclusão	Possíveis Ações										
1	Um mês no gráfico possui um número de issues criadas acima do previsto pela equipe	Meses maiores no gráfico são meses que há maior participação dos colaboradores no desenvolvimento de suas tarefas, como, por exemplo, projetos que utilizam issues para rastreamento de todas as tarefas do projeto	Deve-se tomar ações de incentivo e direcionamento em caso de mês com baixa abertura de issues ou analisar a quantidade de bugs solucionados. Além disso, deve-se investigar os tipos das issues criadas nesses meses e entender com a equipe a razão da criação dessas issues										
2	Um tipo de issue consiste de 50% ou mais das issues de um mês	Um tipo de issue recorrente pode indicar um ciclo de melhorias na aplicação, devido a requisições constantes de novas funcionalidades, bem como pode indicar a presença de erros na aplicação	Ao identificar o tipo de issue recorrente, a equipe deve-se reunir e compreender a razão daquele tipo aparecer muitas vezes. Se for do tipo bug, por exemplo, a equipe deve tomar mais atenção com a fase de testes durante o desenvolvimento a fim de diminuir o número de issues do tipo bug.										



Objetivos

Objetivo Estratégico: Aumentar a eficácia da gestão da produtividade

Objetivo de Medição: Monitorar os commits de código fonte no projeto.

Tipo

controle
 qualidade
 desempenho

MR-MPS-SW NÍVEL: F
 PROC.: GCO

Representação Gráfica do Indicador

Procedimento de Medição e Análise

MEDIÇÃO

INÍCIO: Em qualquer fase do projeto, respeitando a frequência de execução do indicador

Medida: CMA: Coletar dados sobre os commits de código na área Commits do Github através do PyNdicators

Atualizar dados na Planilha de Medição do Indicador

Planilha de Medição Atualizada

ANÁLISE

FIM: Executar o Procedimento de Análise Baseado em Relacionamentos

Avaliar gráfico gerado e executar o Procedimento de Análise Baseado em Critérios disponibilizado a Ficha de Descrição do Indicador

Geração automática do gráfico para a visualização dos commits por Colaborador por mês e ano na planilha de medição (CMA)

Selecionar dados relevantes para a análise. (Especificar o período de medição e/ou colaboradores).

Medida Base: CMA: COMMITS DE CÓDIGO DE CADA COLABORADOR | Periodicidade: MENSAL | Responsáveis: Coleta: GERENTE DO PROJETO | Análise: GERENTE DO PROJETO

Procedimento de Análise Baseado em Relacionamentos

Indicador **CMA**

PCA: Porcentagem de Commits por Autor

IEA: Inclusões x Exclusões por Autor

RCM: Ranking de Colaboradores por Mês

NIC: Número de Issues criadas por mês x ano

O número de entregas por mês / ano, quando relacionado a Porcentagem de Entregas por Autor permite o gestor do projeto analisar a produtividade de cada um dos autores/colaboradores. Um aumento no número de commits resulta em uma porcentagem maior para um determinado colaborador

Relação entre o número de entregas de código por mês / ano e as Inclusões x Exclusões por Autor pode indicar a produtividade e a participação no projeto. Logo, um aumento número de commits pode levar a um aumento, consequentemente, das alterações de linha de código

As entregas acumuladas de cada mês são utilizadas para determinar o colaborador que realizou o maior número de entregas de código fonte a cada mês

Os meses com maior número de issues podem significar um aumento no número de commits de código fonte a fim de solucionar as tarefas atribuídas aos colaboradores

Procedimento de Análise Baseado em Critérios

ID	Premissa	Possível Conclusão	Possíveis Ações
1	O número de commits de um determinado mês corresponde a 20% ou mais dos commits totais em um ano	Meses maiores no gráfico são meses que há maior participação dos colaboradores no desenvolvimento de suas tarefas.	Deve-se tomar ações de incentivo e direcionamento em caso de mês com baixa entrega de código fonte. O gerente deve avaliar as tarefas referente àquele mês e verificar se há entregas pendentes, procurando priorizar o desenvolvimento de módulos com qualidade
2	Um colaborador possui 50% ou mais dos commits dentro de um mês	Isso pode indicar que um colaborador está tendo uma dedicação maior ao projeto que os outros colaboradores	O gerente do projeto deve se reunir com os colaboradores do projeto e entender o motivo do atual grau de participação de cada um

PÁG.: x/xx

81



Objetivos

Objetivo Estratégico: Aumentar a eficácia da gestão do código fonte

Objetivo de Medição: Monitorar os commits de código fonte no projeto.

Tipo: controle qualidade desempenho

MR-MPS-SW

NÍVEIS: F

PROC.: GCO

Representação Gráfica do Indicador

Mês: Mês

Soma de Inclusões Soma de Exclusões

Inclusões x Exclusões por Autor

author	Soma de Inclusões	Soma de Exclusões
benmarwick	1	1
braganholo	19556	22531
JoaoFelipe	143306	57875
leomurta	43448	37136
remram44	10	13

Procedimento de Medição e Análise

MEDIÇÃO

Status atualizado do Projeto

INÍCIO: Em qualquer fase do projeto, respeitando a frequência de execução do indicador

Medida: IEA

Coletar dados sobre as alterações em linhas de código por commit na área Commits do Github através da Aplicação

Atualizar dados na Planilha de Medição do Indicador

Planilha de Medição Atualizada

ANÁLISE

FIM

Executar o Procedimento de Análise Baseado em Relacionamentos

Avaliar gráfico gerado e executar o Procedimento de Análise Baseado em Critérios disponibilizado na Ficha de Descrição do Indicador

Geração automática do gráfico para a visualização das alterações de código por Colaborador na planilha de medição (IEA)

Selecionar dados relevantes para a análise. (Especificar colaboradores).

Procedimento de Análise Baseado em Relacionamentos

Indicador IEA

PCA: Porcentagem de Commits por Autor

Colaboradores com um alto número de commits geralmente possuem um alto número de linhas adicionadas e deletadas. Isso indica um alto grau de participação no projeto. Analogamente, colaboradores que não tem muitas linhas de código no projeto tem poucos commits.

RCM: Ranking Colaborador por Mês

O gerente de projeto pode analisar qual colaborador teve a maior participação dentro de um período de tempo e o número de linhas alteradas serve como uma base para isso, além de indicar quais colaboradores estão realizando um maior número de alterações no projeto

CMA: Commits por Mês x Ano

Relação entre o número de commits de código por mês / ano e as Inclusões x Exclusões por Autor pode indicar a produtividade e a participação no projeto.

Medida Base: IEA: LINHAS DE CODIGO INCLUIDAS EM COMMITS DE CADA COLABORADOR E LINHAS DE CODIGO EXCLUIDAS EM COMMITS DE CADA COLABORADOR

Periodicidade: MENSAL

Responsáveis: Coleta: GERENTE DO PROJETO Análise: GERENTE DO PROJETO

Procedimento de Análise Baseado em Critérios

ID	Premissa	Possível Conclusão	Possíveis Ações
1	Colaborador possui 50% a mais linha de código excluídas do que incluídas	Isso pode ocorrer devido a otimização de código antigo ou correção de códigos de outros colaboradores	Não há ação necessária
2	Colaborador possui 50% a mais linha de código incluídas do que excluídas	Adição de novas funcionalidades ou até mesmo correção de código pode apresentar um aumento no número de linhas adicionadas	
3	Um colaborador é responsável por 50% das alterações de linhas código	Isso pode indicar que um colaborador está tendo uma dedicação maior ao projeto que os outros colaboradores	

O gerente do projeto deve se reunir com os colaboradores do projeto e entender o motivo do atual grau de participação de cada um. Além disso, o gerente pode procurar redistribuir as issues a fim de não sobrecarregar um determinado colaborador do projeto

PÁG.: x/xx

82

