

Uso de Atributos Não-técnicos na Alocação de Equipes em Projetos de Desenvolvimento Distribuído de Software

Bruno Ribeiro¹, Glêdson Elias¹

¹Centro de Informática – Universidade Federal da Paraíba (UFPB) – COMPOSE
João Pessoa – PB – Brasil

bruno@compose.ufpb.br, gledson@di.ufpb.br

Abstract. *Taking into account the benefits of global software development, many organizations allocate implementation tasks among globally distributed teams. However, their geographical distance generates several issues which reduce the communication effectiveness, directly affecting the project's progress. In order to mitigate such issues, based on non-technical features of distributed development teams and dependences among software modules, this paper describes an approach for recommending the allocation of development teams to implementation tasks of software modules, detailing the used attributes on the non-technical analysis and how they are obtained.*

Resumo. *Levando em consideração os benefícios do desenvolvimento distribuído de software, muitas organizações alocam atividades de implementação para equipes globalmente distribuídas. Entretanto, sua distância geográfica gera uma série de problemas que reduzem a efetividade da comunicação, afetando diretamente o progresso do projeto. Com o objetivo de reduzir esses problemas, baseando-se nos aspectos não-técnicos das equipes distribuídas e nas dependências entre módulos de software, este artigo descreve uma abordagem para alocação de equipes de desenvolvimento para módulos de software, detalhando os atributos utilizados na análise não-técnica e como eles são obtidos.*

1. Introdução

Com a globalização, a indústria busca novos mecanismos para obter vantagens de mercado e, na indústria de software, não é diferente. A adoção de processos de Desenvolvimento Distribuído de Software (DDS) vem crescendo significativamente, fazendo com que o desenvolvimento de software passe a acontecer dividido em diferentes pontos geográficos, seja ele com equipes subsidiárias da própria organização (*insourcing*) ou equipes terceirizadas (*outsourcing*). Com a utilização de DDS, os benefícios esperados incluem a redução do *time-to-market*, a melhoria da qualidade, a redução de custos e a maior qualidade de mão de obra (Audy e Prikladnicki, 2007).

Além disso, o fator humano, segundo a norma NBR ISO 10006 (ABNT, 2000), é um fator chave para determinar a qualidade de um projeto. No DDS, esses fatores tornam-se ainda mais sensíveis, uma vez que os envolvidos estão separados geograficamente, o que gera dificuldades temporais, culturais e de afinidade entre as equipes (Herbsleb e Mockus, 2003) (Audy e Prikladnicki, 2007) (Gumm, 2006).

Dado que existem diversas equipes distribuídas globalmente, processos de alocação são utilizados com o objetivo de selecionar as equipes que virão a participar de um projeto. Porém, a maior parte deles leva em consideração apenas atributos técnicos, como conhecimentos, habilidades e experiências em uma determinada tecnologia

(Duggan, Byrne e Lyons, 2004) (Silva, 2007) (Callegari, Foliatti e Bastos, 2009). Os aspectos não-técnicos são reduzidos a uma verificação de histórico ou afinidade, o que é insuficiente para a complexidade dos problemas decorrentes da utilização de DDS.

Neste contexto, a fim de mitigar os problemas de comunicação enfrentados pela dispersão geográfica causada pela adoção de DDS, este artigo apresenta uma abordagem para alocação de equipes distribuídas para atividades de implementação de módulos de software, onde são analisados aspectos não-técnicos das equipes e a dependência entre os módulos de software. Como premissa para avaliar o grau de comunicação demandado entre as equipes, a abordagem considera o grau de dependência existente entre os módulos de software e seus componentes constituintes, já que, segundo (Ghezzi, Jazayeri e Mandrioli, 2002), a dependência dos componentes influencia na comunicação requerida entre suas respectivas equipes de desenvolvimento.

A abordagem aqui proposta corresponde a terceira fase de um framework de recomendações, inicialmente proposto em (Pereira, Santos, *et al.*, 2010). Neste framework, a primeira fase identifica e mensura o grau de dependência entre os módulos que compõem uma linha de produto de software. Em seguida, a segunda fase identifica e ranqueia as equipes de desenvolvimento tecnicamente qualificadas para implementá-los. Por fim, a terceira fase, detalhada neste artigo, recomenda a alocação de equipes para implementação dos módulos.

As próximas seções estão organizadas da seguinte forma. A seção 2 apresenta sucintamente o framework de recomendações, do qual a abordagem aqui proposta é parte constituinte. A Seção 3 apresenta a abordagem proposta. A seção 4 apresenta os trabalhos relacionados. E, por fim, a seção 5 apresenta as considerações finais.

2. Framework de Recomendação

Como já mencionado, o presente trabalho está contido em um *framework* de recomendações para alocação de equipes em projetos de desenvolvimento distribuído de Linhas de Produto de Software (LPS), conceitualmente proposto em (Pereira, Santos, *et al.*, 2010). O *framework* tem o propósito de reduzir os problemas de comunicação enfrentados durante a fase de implementação de uma LPS, onde as equipes de desenvolvimento estão globalmente distribuídas. O *framework* é composto de três fases: *recomendação de módulos*, *recomendação de equipes hábeis*, e *recomendação de alocação das equipes*. Vale ressaltar que, após estas três fases, o gerente de projeto, juntamente com as equipes participantes podem realizar a avaliação dos trabalhos realizados, retroalimentando os atributos das equipes para novos projetos.

A primeira fase, denominada *recomendação de módulos*, tem o objetivo de agrupar componentes relacionados que compõem a arquitetura de software em unidades de alocação e implementação, aqui denominadas *módulos de software*. Considerando que o grau de relacionamento entre os componentes interfere no grau de comunicação das equipes que os implementam (Ghezzi, Jazayeri e Mandrioli, 2002), esta fase agrupa os componentes fortemente dependentes, reduzindo a dependência externa aos módulos e aumentando a dependência interna aos módulos. Desta forma, alocando cada módulo a uma determinada equipe, é esperada uma redução no grau de comunicação requerido para implementá-los. Como resultado, essa fase recomenda um conjunto de módulos de software, indicando o grau de dependência existente entre os mesmos.

A segunda fase, denominada *recomendação de equipes hábeis*, tem o propósito de identificar as equipes que possuem habilidades técnicas para implementar os módulos de software. Essa fase calcula uma métrica de adequabilidade técnica, que

sinaliza o quão indicada cada equipe é para implementar cada módulo de software. A adequabilidade leva em consideração políticas de alocação escolhidas pelo gerente de projeto, que podem incluir a redução de custos ou a melhoria do *time-to-market*. Como resultado, essa fase recomenda um conjunto de equipes para cada módulo de software. Vale ressaltar que as equipes são ranqueadas de acordo com a adequabilidade.

Enquanto a segunda fase avalia os aspectos técnicos das equipes, a terceira fase, denominada *recomendação de alocação de equipes*, avalia os aspectos não técnicos das equipes. O propósito desta fase é avaliar a viabilidade de comunicação efetiva entre as equipes tecnicamente habilitadas para implementar cada módulo de software. Como resultado, esta fase recomenda um conjunto de soluções de alocação de equipes a módulos de software, ranqueadas de acordo com uma métrica de viabilidade de comunicação. Como já mencionado, a principal contribuição deste artigo é detalhar esta terceira fase do *framework* conceitual, o que será realizado na próxima sessão.

3. Uma abordagem para recomendação de alocação de equipes

A alocação de equipes é realizada com base em atributos não-técnicos. É utilizado como entrada das duas fases anteriores do *framework* de recomendações dois artefatos, como pode ser visto na Figura 1. A recomendação de módulos oferece como entrada o artefato MBM (*Module Based Matrix*), onde é possível identificar os relacionamentos externos e os relacionamentos internos aos módulos. E a recomendação de equipes hábeis oferece o TMM (*Team x Module Mapping*), ao qual identifica, para cada módulo, a lista de equipes hábeis e ranqueadas de acordo com a adequabilidade técnica.

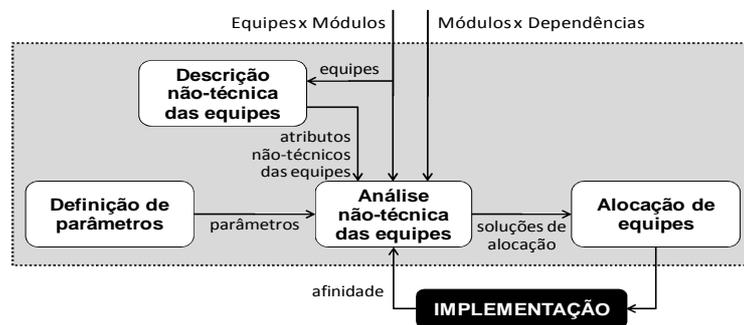


Figura 1. Recomendação de alocação

A recomendação de alocação é dividida em duas etapas, a descrição não-técnica, que identifica os atributos que causam impacto na comunicação, e a análise de alocação, que é composta pela configuração, análise não-técnica e a seleção de recomendação.

3.1. Descrição não-técnica das equipes

A descrição não-técnica das equipes tem como objetivo caracterizar informações das equipes que, a partir das mesmas, seja possível prever a viabilidade e a qualidade de comunicação entre as equipes geograficamente distribuídas.

Os atributos não-técnicos foram fundamentados em discussões e sugestões encontradas em diversos trabalhos da literatura (Warren McFarlan, 1996), (Gumm, 2006), (Audy e Prikladnicki, 2007) e (Lamersdorf e Münch, 2010). Sendo assim, problemas como a alta distância geográfica, diferentes horários de trabalho, diferentes padrões organizacionais e infra-estrutura, falta de afinidade e idiomas conflitantes podem ser analisados sob o ponto de vista de três tipos de atributos: **temporal** (*fuso-horário, horário de trabalho, dias semanais de trabalho*), **cultural** (*idiomas conhecidos*

e perfil organizacional e de infra-estrutura) e de **afinidade** (qualidade da comunicação).

A obtenção dos atributos de cada equipe é feita através de questionários que podem ser destinados ao gerente da equipe de desenvolvimento ou a todos os membros da equipe. Alguns atributos, chamados atributos diretos, são obtidos pela resposta direta do questionário, enquanto que outros atributos, atributos indiretos, precisam passar por funções que calculam um valor matemático do atributo para a equipe. Esse cálculo é feito por existirem várias características que devem ser analisadas, ou por ser necessário obter informação de vários membros para identificar o *expertise* da equipe.

Os atributos temporais são atributos diretos, já os atributos culturais e de afinidade necessitam ser derivados. Para os culturais, para identificar os idiomas, um questionário é respondido por todos os membros da equipe, sobre os critérios de: *escrita, leitura, fala e entendimento* (respectivamente *e, l, f, en*). Cada um deles pode se enquadrar nos níveis *1-baixo, 2-mediano, 3-ótimo e 4-nativo*. A avaliação de um membro no idioma é definido pela média dos níveis fornecidos. Caso algum membro não possua conhecimento em um determinado idioma, ele será assumido como valor 0. Para se manter no intervalo [0,1], a seguinte equação é utilizada:

$$a_{id}(e_i) = \left(\frac{\sum_{i=1}^{i=n} (a_{e,id}(m_n) + a_{l,id}(m_n) + a_{f,id}(m_n) + a_{en,id}(m_n))}{n * 4 * 4} \right) \quad (1)$$

O perfil organizacional e de infra-estrutura também é um atributo indireto, identificado pelas respostas do gerente da equipe a um questionário que identifica características organizacionais e de infra-estrutura, semelhantes ao encontrado no modelo exibido na Figura 2. Com base nas respostas, o perfil é criado com um valor entre [0, 1], onde o 0 identifica uma equipe com fraca organização e infra-estrutura, enquanto que o 1 identifica uma equipe com alta organização e infra-estrutura. O questionário deve conter perguntas sobre processos de software, processo de maturidade, infra-estrutura para ferramentas colaborativas e com que frequência essas ferramentas podem ser utilizadas.

<p>A equipe é dedicada a se adaptar a novos processos de software?</p> <p><input type="checkbox"/> Nenhum: não utiliza processos de software</p> <p><input type="checkbox"/> Baixo: utiliza apenas os processos internos</p> <p><input type="checkbox"/> Mediano: disponível para se adaptar a novos processos de simples adequação</p> <p><input type="checkbox"/> Bom: disponível para se adaptar a qualquer processo convencional</p> <p><input type="checkbox"/> Ótimo: disponível para se adaptar a processos de LPS</p> <p>Qual o nível da equipe em modelos de maturidade?</p> <p><input type="checkbox"/> Não utiliza modelo de maturidade</p> <p><input type="checkbox"/> Baixo – flexível para utilizar níveis baixos de modelos</p> <p><input type="checkbox"/> Mediano – flexível para utilizar níveis intermediários dos modelos</p> <p><input type="checkbox"/> Alto – flexível apenas para os níveis mais altos dos modelos</p> <p>Nível de ferramentas colaborativas disponíveis:</p> <p><input type="checkbox"/> Baixo – Email e mensageiro instantâneo</p> <p><input type="checkbox"/> Mediano – Audioconferência utilizando VoIP ou linhas telefônicas</p> <p><input type="checkbox"/> Bom – Videoconferência desktop (um para um)</p> <p><input type="checkbox"/> Ótimo – Videoconferência multicast (n para n)</p> <p>No caso de videoconferência, a disponibilidade das ferramentas é integral?</p> <p><input type="checkbox"/> Não utilizo <input type="checkbox"/> Parcial: depende de verificação de disponibilidade <input type="checkbox"/> Total</p>
--

Figura 2. Questionário modelo para perfil organizacional

Para se definir um valor numérico baseado nas respostas, a teoria dos conjuntos da lógica fuzzy é utilizada da seguinte forma:

$$a_{po}(e_i) = \frac{\sum_{perg} \left(\frac{resultado}{|alternativas|} \right)}{|perg|} \quad (2)$$

Sabendo que no caso de uma pergunta em que a opção “nenhum” é possível, o valor de *resultado* e de *alternativas* deve ser subtraído por 1.

Por fim, a afinidade entre as equipes, que também é um valor numérico entre [0,1], possui valor inicial 0,5 e sofre alteração com base na retroalimentação realizada pelos gerentes das equipes. Ao final de cada projeto, após a implementação dos módulos, os gerentes das equipes avaliam a eficácia e efetividade da comunicação com as equipes relacionadas. A retroalimentação pode ser realizada de duas formas, utilizando avaliação incremental ou avaliação absoluta. A primeira delas é utilizada de forma a alterar gradativamente a afinidade entre as duas equipes de acordo com o feedback do gerente. Seguindo também a teoria de conjuntos de fuzzy, é escolhido um valor dentre os quatro valores de satisfação: *completamente insatisfeito*, *levemente insatisfeito*, *indiferente*, *levemente satisfeito* e *totalmente satisfeito*. Os valores numéricos identificados são respectivamente {0}, {0,25}, {0,5}, {0,75} e {1}. O novo valor da afinidade é derivado a partir da seguinte função, sabendo que w_{a1} e w_{af} são pesos que incrementam e decrementam a importância da avaliação atual e o feedback, respectivamente:

$$a_a(e_i, e_j) = \left(\frac{w_{a1} \times a_a(e_i, e_j) + w_{af} \times feedback_{ei}(e_j)}{w_{a1} + w_{af}} \right) \quad (3)$$

A avaliação absoluta pode ser utilizada como alternativa a avaliação incremental e tem o objetivo de oferecer um feedback irrestrito, modificando a afinidade de forma não-incremental, para os valores citados.

3.2. Análise de alocação

O primeiro passo da análise de alocação é a definição de parâmetros. O impacto causado pelas características não-técnicas terão implicação nas dimensões temporal, cultural e de afinidade. Nessa etapa, as três dimensões deverão ter pesos definidos, identificando a prioridade que o gerente de projeto tem como objetivo. A não definição dessa prioridade manterá os pesos iguais. Além disso, os pesos utilizados na afinidade também devem ser definidos.

A segunda etapa é a análise não-técnica. Nessa análise, o requisito que estima o nível de comunicação almejado é a proporcionalidade com o relacionamento externo entre os módulos de software. Quanto maior for o relacionamento, maior deverá ser a avaliação não-técnica entre as equipes.

Pela complexidade do problema e o universo de possibilidades ser demasiadamente grande, foi utilizado a técnica de algoritmos genéticos como algoritmo de meta-heurística. A função objetivo, que define o quão bom é uma solução, utiliza como critério uma média ponderada com base nas três avaliações: avaliação temporal, cultural e de afinidade, além dos pesos definidos pelo gerente.

Dada uma solução de alocação s , as avaliações temporal, cultural e de afinidade resultam em valores no intervalo [0, 1], representadas pelos termos $A_t(s)$, $A_c(s)$ e $A_a(s)$, respectivamente. Com o propósito de comparar a qualidade das soluções de alocação candidatas, estes termos são então aplicados a função-objetivo $f(s)$ abaixo:

$$f(s) = \left(\frac{w_t \times A_t(s) + w_c \times A_c(s) + w_a \times A_a(s)}{w_t + w_c + w_a} \right) \quad (4)$$

Os valores de w_t , w_c e w_a representam os pesos definidos pelo gerente de projeto para as dimensões temporal, cultural e de afinidade, respectivamente.

Para uma solução s , a avaliação em uma dada dimensão x é calculada como uma média ponderada do nível de dependência entre cada par de módulos (m_r, m_s) e o valor da avaliação naquela dimensão entre as equipes (e_i, e_j) alocadas ao par de módulos (m_r, m_s). Formalmente, as avaliações temporal, cultural e de afinidade são calculadas usando uma expressão similar à indicada na abaixo, porém substituindo os termos genéricos $A_x(s)$ e $a_x(e_i, e_j)$ por $A_t(s)$ e $a_t(e_i, e_j)$, $A_c(s)$ e $a_c(e_i, e_j)$, e $A_a(s)$ e $a_a(e_i, e_j)$, respectivamente. Note que, nesta equação, o termo $d(m_r, m_s)$ representa o nível de dependência entre os módulos i e j . Já o termo $a_x(e_i, e_j)$ representa o valor da avaliação da respectiva dimensão x entre o par de equipes alocadas aos módulos i e j .

$$A_x(s) = \frac{\sum_{r \neq s} d(m_r, m_s) \times a_x(e_i, e_j)}{\sum_{r \neq s} d(m_r, m_s)} \quad (5)$$

Os valores das avaliações temporal $a_t(e_i, e_j)$, cultural $a_c(e_i, e_j)$ e de afinidade $a_a(e_i, e_j)$ entre o par de equipes alocado aos módulos i e j são mensuradas levando-se em consideração os atributos temporais, culturais e de afinidade destas equipes. A avaliação temporal é calculada, conforme indicado na Equação (6). Observe que, o lado esquerdo da equação representa o percentual de horas de trabalho semanal que as equipes e_i e e_j compartilham simultaneamente em relação ao conjunto total de horas de trabalho semanal de todas as equipes candidatas. Já o lado direito da equação representa um fator de penalidade proporcional ao esforço de comunicação requerido das equipes e_i e e_j para implementar todos os módulos aos quais estão alocados. Nesta porção da equação, o termo $d(m, m)$ representa a dependência interna do módulo m , e M_{e_i} representa o conjunto de módulos alocado à equipe e_i . Note que, quanto maior o número de módulos alocados as equipes, maior o esforço requerido, e, assim, menor a disponibilidade das equipes para comunicação com outras equipes.

$$a_t(e_i, e_j) = \frac{|wt(e_i) \cap wt(e_j)|}{|U_k wt(e_k)|} \times \left(1 - \frac{\sum_{m \in M_{e_i}} d(m, m) + \sum_{m \in M_{e_j}} d(m, m)}{2 \times \sum_m d(m, m)} \right) \quad (6)$$

Por sua vez, como indicado, a avaliação cultural para um par de equipes é calculada como a média entre a avaliação do melhor idioma comum às equipes $a_{id}(e_i, e_j)$ e da avaliação do perfil organizacional $a_{org}(e_i, e_j)$ das equipes. A princípio, o cálculo do valor da avaliação do melhor idioma $a_{id}(e_i, e_j)$ e do perfil organizacional $a_{org}(e_i, e_j)$ pode ser realizado como a média dos níveis de idioma e dos níveis de perfil organizacional de cada equipe, respectivamente. No entanto, a distância entre os níveis de idioma e de perfil organizacional também é importante (Lamersdorf e Münch, 2010). Desta forma, $a_{id}(e_i, e_j)$ e $a_{org}(e_i, e_j)$ são calculados conforme indicado na Equação (8), onde os termos $\mu_x(e_i, e_j)$ e $\sigma_x(e_i, e_j)$ representam, respectivamente, a média e o desvio padrão do atributo x (nível de idioma ou perfil organizacional) das equipes. Logo, quanto mais distantes os níveis de idioma ou perfil organizacional, maior a influência negativa na avaliação do respectivo atributo.

$$a_x(e_i, e_j) = \mu_x(e_i, e_j) \times (1 - \sigma_x(e_i, e_j)) \quad (8)$$

No caso da avaliação da afinidade entre um par de equipes $a_a(e_i, e_j)$, o valor também é calculado considerando a média e o desvio padrão entre os níveis de afinidade $a_{i \rightarrow j}$ e $a_{j \rightarrow i}$, de forma análoga a Equação (8).

Ao final da execução do algoritmo, as melhores soluções serão mostradas ao gerente de projetos, levando em consideração a função objetivo. As informações fornecidas para o gerente sobre as soluções são o seu ranqueamento, o valor da função objetivo, as três avaliações não-técnicas (temporal, cultural e afinidade), além da alocação de cada solução. Dessa forma, o gerente tem como usar sua experiência na seleção da alocação.

4. Trabalhos Relacionados

Esta sessão compara a abordagem proposta com alguns trabalhos de alocação de recursos, evidenciando suas principais características e os pontos em que diferem. Em (Barreto, 2005) é apresentado uma abordagem para a alocação de recursos humanos baseado em satisfação de restrições. Esta abordagem procura por uma solução válida baseada nas restrições definidas. De forma similar a proposta aqui apresentada, esta abordagem avalia aspectos técnicos e não-técnicos dos profissionais. No entanto, ao contrário da atual abordagem, não considera aspectos culturais e de afinidade, apenas aspectos temporais. Outro diferencial é o fato desta abordagem alocar pessoas a atividades do projeto, enquanto a presente aloca equipes distribuídas, considerando as características técnicas e não-técnicas individuais dos profissionais que as compõem.

Duggan, Byrne e Lyons (Duggan, Byrne e Lyons, 2004) propõe um mecanismo de otimização baseado em algoritmo genético para alocação de tarefas na fase de construção do software. No entanto, este mecanismo realiza apenas a análise de aspectos técnicos das equipes, levando em conta o nível de conhecimento nos seguintes assuntos: segurança, rede, base de dados, controle e interface de usuário. Em contraste, a abordagem aqui proposta, juntamente com o framework de recomendação do qual é parte integrante, consideram tanto os aspectos técnicos quanto os aspectos não-técnicos.

Lamersdorf e Munch (Lamersdorf e Münch, 2010) apresentaram um método baseada em redes Bayesianas para alocar tarefas a equipes distribuídas. Neste método, aspectos técnicos e não-técnicos das equipes são também considerados, por exemplo, o custo para realizar as atividades e o custo de comunicação entre as localidades. Embora considere aspectos não-técnicos das equipes, esta ferramenta não define mecanismos para coletar informações sobre tais aspectos não-técnicos. Ao contrário da abordagem aqui proposta, cuja coleta de informações é baseada em questionários submetidos aos gerentes de equipes e seus membros. Este método adota apenas tabelas probabilísticas, não refletindo, portanto, a realidade e as especificidades de cada equipe

Apesar dos avanços, apenas uma das propostas apresentadas lida com projetos de DDS nos quais as equipes de desenvolvimento estão geograficamente dispersas, analisando, na maior parte, apenas habilidades técnicas das equipes. No entanto, no contexto de DDS, como já mencionado, os aspectos não técnicos interferem consideravelmente no progresso e sucesso de um projeto de software. Nesta direção, a abordagem proposta neste artigo considera os aspectos não técnicos das equipes.

5. Considerações Finais

A alocação de equipes é uma atividade muito complexa para ser realizada apenas baseada na experiência e conhecimento do gerente de projetos. Devido à quantidade de possíveis soluções, a tomada de decisão de forma *ad hoc* é ineficiente e incompleta. Embora já existam na literatura algumas propostas para sistematizar a alocação de equipes, aspectos específicos de projetos de DDS não são considerados.

Desta forma, a principal contribuição deste trabalho é apresentar uma abordagem sistemática para alocação de equipes em projetos de DDS, considerando os aspectos não-técnicos das equipes envolvidas. A abordagem não substitui ou despreza a experiência e conhecimento do gerente de projetos. Porém, representa uma opção mais efetiva e eficiente de obter um conjunto de soluções com bom desempenho das equipes em termos de viabilidade de comunicação, deixando sob a responsabilidade do gerente de projeto a seleção da solução a ser adotada, considerando sua experiência e conhecimento, bem como as especificidades do projeto de DDS em questão.

Atualmente, um estudo de caso está sendo realizado com o propósito de validar os resultados obtidos. Além disso, o desenvolvimento de ferramentas para automatização da abordagem também está em andamento. Como limitação atual da abordagem, a proposta ainda não considera o aspecto de custo do projeto, uma vez que estão sendo priorizados os estudos que identifiquem quais os atributos mais importantes para efetivar a comunicação entre as equipes.

Referências

- ABNT, N. I. 1. Gestão da Qualidade – Diretrizes para a Qualidade no Gerenciamento de Projetos. Rio de Janeiro, RJ Brasil: Associação Brasileira de Normas Técnicas, 2000.
- AUDY, J.; PRIKLADNICKI, R. Desenvolvimento distribuído de software. Rio de Janeiro: Campos/Elsevier, 2007.
- BARRETO, A. Apoio à alocação de recursos humanos em projetos de software: uma abordagem baseada em satisfação de restrições. Thesis, 2005.
- CALLEGARI, D. A.; FOLIATTI, F. L.; BASTOS, R. M. MRES - Ferramenta para seleção de recursos para tarefas de projetos de software via abordagem difusa e multicritérios. SBES. Fortaleza: [s.n.]. 2009.
- DUGGAN, J.; BYRNE, J.; LYONS, G. J. A Task Allocation Optimizer for Software Construction. IEEE Software. vol. 21, issue 3. [S.l.]: [s.n.]. 2004. p. 76-82.
- GHEZZI, C.; JAZAYERI, M.; MANDRIOLI, D. Fundamentals of Software Engineering. Upper Saddle River: Prentice Hall, 2002.
- GUMM, D. C. Distribution Dimensions in Software Development Projects: A Taxonomy. IEEE Software, vol3, issue 5, September & October 2006. 45-51.
- HERBSLEB, J. D.; MOCKUS, A. An empirical study of speed and communication in globally-distributed software development. IEEE Trans. Softw. End. vol. 29, no. 6, 2003. 481-494.
- LAMERSDORF, A.; MÜNCH, J. Studying the Impact of Global Software Development Characteristics on Project Goals: A Causal Model. The Open Software Engineering Journal. [S.l.]: [s.n.]. 2010. p. pg 2-13.
- LINDEN, R. Algoritmos Genéticos. Rio de Janeiro: Brasport, 2008.
- PEREIRA, T. et al. A Recommendation Framework for Allocating Global Software Teams in Software Product Line Projects. Second International Workshop on Recommendation Systems for Software Engineering (RSSE 2010), 2010.
- SILVA, M. A. WebAPSEE-Planner: Auxílio à Alocação de Pessoas em Projetos de Software através de políticas. UFPA. [S.l.]. 2007.
- WARREN MCFARLAN, F. Issues in Global Outsourcing. Global Information Technology and Systems Management. Nashua: Ivy League Publishing. 1996.