

HoneyComb-ER: Adequação do Framework HoneyComb para Engenharia de Requisitos

Lis Ângela De Bortoli, Emerson Rogério de Oliveira Junior

Instituto Federal do Rio Grande do Sul (IFRS) - Campus Sertão - Rodovia RS 135,
Km 25 - Distrito Eng. Luiz Englert - CEP: 99170-000 - Sertão-RS

{lis.debortoli,emerson.oliveira}@sertao.ifrs.edu.br

***Abstract.** Requirements Engineering (RE) includes people with different backgrounds and cultures. It is necessary to observe RE as a social structure, which involves intense communication. The use of software and social networks in RE was boosted by the COVID-19 pandemic, with the rise of remote work as they allow interaction between people in a flexible and horizontal way. The HoneyComb framework represents the fundamental elements of social softwares. The objective is to propose the HoneyComb-ER adequacy, which encompasses specific elements of the RE supporting the development of social software in this area. As a result there is a structure of nine elements based on the bibliographic and comparative analysis of specialists.*

***Resumo.** A Engenharia de Requisitos (ER) inclui pessoas com diferentes conhecimentos e culturas. É preciso observar a ER como uma estrutura social, que envolve intensa comunicação. O uso de softwares e redes sociais na ER foi impulsionado pela pandemia de COVID-19, com a ascensão do trabalho remoto, pois permitem a interação entre pessoas de forma flexível e horizontal. O framework HoneyComb representa os elementos fundamentais de softwares sociais. O objetivo é propor o HoneyComb-ER, adequação que engloba elementos específicos da ER, embasando o desenvolvimento de softwares sociais nesta área. Como resultado, tem-se uma estrutura de nove elementos definida com base na análise bibliográfica e comparativa de especialistas.*

1. Introdução

Independente do processo utilizado, da complexidade ou do tamanho do sistema, a Engenharia de Requisitos (ER), é uma etapa essencial no desenvolvimento de software, cujo objetivo é determinar as funções, as condições e as restrições sob as quais o software irá funcionar, ou seja, os requisitos (Sommerville, 2011). A fim de estabelecer os requisitos de forma completa, correta e não ambígua, faz-se necessário considerar, além dos aspectos técnicos, o ambiente no qual o software será utilizado.

Neste contexto, é possível perceber que a ER possui um viés entre os aspectos técnicos e os aspectos sociais. De um lado estão questões relacionadas à utilização de métodos, técnicas e ferramentas adequados para que o software possa funcionar corretamente, atendendo aos objetivos do cliente e às necessidades técnicas disponíveis. Por outro lado, na maioria das vezes, os requisitos são descobertos com base nas informações extraídas de um conjunto de pessoas que detêm o conhecimento sobre o sistema a ser desenvolvido. Com culturas e conhecimentos diferentes, as pessoas

envolvidas (*stakeholders*), irão interagir para negociar os requisitos, caracterizando o aspecto social.

Atualmente, através da web, informações de qualquer tipo podem ser obtidas de forma simultânea ao acontecimento no mundo real e sob diversas formas: textos, vídeos, áudios, imagens etc. Além disso, ferramentas baseadas em inteligência artificial têm despontado como facilitadoras na obtenção de informações. Com o surgimento das redes sociais virtuais, foi proporcionado um espaço crescente de comunicação, de forma onipresente, no qual é possível a construção de conhecimento de maneira coletiva e colaborativa. Sendo assim, entende-se rede social como sendo uma estrutura social composta por pessoas e/ou organizações, conectadas por um ou vários tipos de relações, que partilham valores e objetivos comuns (Marteletto, 2010). As redes sociais virtuais possibilitam relacionamentos horizontais e não hierárquicos entre os participantes.

Durante a pandemia de COVID-19, organizações tiveram que se adaptar rapidamente para permitir que seus colaboradores exercessem suas funções remotamente. Isso levou a um aumento na demanda por ferramentas de colaboração online, plataformas de gerenciamento de projetos, comunicação em tempo real, entre outras. Considerando esse cenário, as atividades da ER necessitaram passar por adaptações. Por exemplo, práticas para engenharia de requisitos em equipes remotas de *startups* de software são apontadas pelo estudo de Hooshyar *et al.* (2023) como uma atividade que exige mais atenção, ou seja, em que as práticas devem ser mais desenvolvidas. Eles verificaram também que, em geral, nas equipes de desenvolvimento em *startups* de software, houve um impacto negativo na coleta de requisitos com o trabalho remoto.

Devido ao isolamento social imposto pela pandemia, as relações precisaram ser ressignificadas e as redes sociais passaram a ter papel importante, tanto no campo pessoal quanto no profissional. Utilizadas em diversas áreas, as redes sociais virtuais são classificadas como sendo softwares sociais, ou seja, ferramentas que enfatizam a colaboração entre pessoas e que possuem mecanismos que facilitam a conexão social e a troca de informações (Martino, 2015). Smith (2007) criou o *framework* HoneyComb, que contém uma estrutura de referência para o desenvolvimento de softwares sociais. A estrutura, que conta com sete elementos, representa os aspectos fundamentais que todo software social deve possuir.

A fim de facilitar o desenvolvimento de ferramentas que apoiam a ER e que tenham como base conceitos e características de softwares sociais, este trabalho tem por objetivo discutir os aspectos sociais da ER e propor uma adaptação do HoneyComb. Pautou-se na análise bibliográfica e comparativa entre os elementos do *framework* HoneyComb e das características e necessidades da ER.

A seção 2 deste artigo aborda conceitos sobre a ER, enfocando o aspecto social. Na seção 3 são enfatizados os princípios fundamentais das redes e softwares sociais, bem como o *framework* HoneyComb. Na seção 4, o *framework* HoneyComb-ER é apresentado e, por fim, as considerações finais e as referências utilizadas.

2. Aspectos sociais da ER

Para Boehm e Egyed (1998), ER é a área para desenvolver uma especificação completa, consistente e não ambígua, que sirva como base para um acordo entre todas as partes envolvidas, descrevendo o que o produto software irá fazer. Assim, entende-se que a ER

é a subárea da engenharia de software que procura sistematizar o processo de definição de requisitos. Sommerville e Sawyer (1997) indicam a ER como sendo o processo de descobrir, analisar, documentar e verificar esses serviços e restrições.

Leite e Doorn (2003) conceituam ER como sendo uma área de pesquisa que procura atacar um ponto fundamental no processo de produção de um software que é a definição do que se quer produzir. Cabe à ER propor métodos, técnicas e ferramentas que facilitem a definição do que se quer de um software. Na visão de Pressman (2016), a ER ajuda os engenheiros de software a compreenderem melhor o problema que eles vão trabalhar para resolver, incluindo o conjunto de tarefas que levam a um entendimento de qual será o impacto do software sobre o negócio, do que o cliente deseja e de como os usuários finais vão interagir com o software. Evidenciam-se, nesses conceitos, os aspectos técnicos da ER. No entanto, é possível identificar que é uma atividade realizada em grupo, em um ambiente específico, que envolve um conjunto de pessoas que têm relação com o software que está sendo construído, revelando o caráter social. Pessoas com diferentes perfis e bagagens culturais se envolvem em busca de um objetivo comum, contribuindo para que o software a ser desenvolvido possa atender às necessidades do ambiente que apoiará.

Neste contexto, vários problemas podem surgir, tais como o conhecimento tácito, a omissão de informações, o conflito de interesses, o envolvimento de indivíduos de diferentes níveis hierárquicos, a resistência a mudanças e a falta de tempo por parte das pessoas do domínio da aplicação para atender a equipe de desenvolvimento. Percebe-se que a maioria dos problemas mencionados são advindos do contato entre pessoas e está relacionado às que têm envolvimento direto ou indireto com o software que está sendo desenvolvido. Além disso, o funcionamento das organizações é bastante complexo e dinâmico e, muitas vezes, não documentado. Assim, despontam as pessoas que atuam nas organizações como sendo as principais fontes de informação.

De acordo com Yu (2009), a ER serve como uma interface entre o mundo do domínio da aplicação e o mundo da tecnologia. Ainda conforme o autor, é importante reconhecer que os requisitos devem ser definidos em conformidade com o ambiente onde o software irá funcionar. Os ambientes envolvem muitos atores sociais com interesses distintos, grande variedade de papéis sendo desempenhados e complexas redes de relacionamentos. Na mesma linha, para Jirotko e Goguen (1994), a área de ER inclui aspectos da Psicologia, da Sociologia e da Ciência da Computação, visto que abrange pessoas com diversos conhecimentos e culturas.

O modelo i^* (Yu, 1997) foi uma tentativa de introduzir alguns aspectos de modelagem social e de raciocínio em métodos de engenharia de sistemas de informação, especialmente em nível de requisitos. Ao contrário dos métodos tradicionais de análise de sistemas, que se esforçam para abstrair os aspectos humanos dos sistemas, i^* reconhece a prioridade dos atores sociais. Os atores são vistos como intencionais, ou seja, têm objetivos, concepções, habilidades e compromissos. A análise foca em quão bem os objetivos de vários atores são alcançados, dada a configuração das relações entre os seres humanos e os atores do sistema e que reconfigurações desses relacionamentos podem ajudar os atores a promoverem seus interesses estratégicos. A estrutura do i^* estimulou consideráveis interesses em uma abordagem socialmente motivada para modelagem e projeto de sistemas, e obteve várias extensões e adaptações.

Posteriormente, os modelos ágeis surgiram como uma tentativa de envolver, de

forma mais próxima, os clientes no processo de desenvolvimento. Assim, é possível realizar entregas mais frequentes e obter um *feedback* rápido sobre a evolução dos requisitos (Sommerville, 2011). Essa categoria de métodos, que tem como alguns de seus representantes, o XP (*eXtreme Programming*), o Scrum, o FDD (*Feature Driven Development*) e o ASD (*Adaptive Software Development*), prevê também a otimização da documentação e uma maior comunicação entre a equipe de desenvolvimento.

Um dos fatores de sucesso dessa abordagem é o cliente dispor de tempo e conseguir permanecer junto à equipe de desenvolvimento. É bastante comum os representantes dos clientes estarem sujeitos a pressões e não poderem participar plenamente do processo. Além disso, pode ser difícil formar uma equipe com talentos e habilidades necessários ao intenso envolvimento, o que pode levar os membros do grupo a não interagirem adequadamente. Cabe salientar que, assegurar o envolvimento do cliente após a entrega do sistema, ainda é um desafio. Segundo Sommerville (2011), métodos ágeis dependem de os membros da equipe compreenderem aspectos do sistema sem consultar a documentação. Se uma equipe é alterada, o conhecimento implícito é perdido e é difícil para os novos integrantes construir o mesmo entendimento do sistema, despontando a dependência das pessoas no processo.

Existem várias ferramentas proprietárias para gerenciamento de requisitos, dentre elas: Helix RM (Perforce), Jira (Atlassian), Orcanos (Application Lifecycle Management) e ReQtest (ReQtest). Por sua vez, as ferramentas OSRMT, SAGERAR e OpenReq são exemplos *open source*. No entanto, a maioria delas aborda os aspectos técnicos, deixando de lado os aspectos sociais.

3. Redes sociais e softwares sociais

Redes sociais podem ser compreendidas como um tipo de relação entre pessoas baseada em flexibilidade e desprovida de hierarquia rígida (caráter horizontal). As redes sociais existem há algum tempo, porém têm recebido mais atenção nas últimas décadas, em função da internet. Sendo assim, o uso das redes sociais tem crescido como ferramenta de comunicação e de construção de conhecimento. Marteleto (2010) define redes sociais como um conceito onipresente e que ocupa espaço crescente no discurso acadêmico, nas mídias, nas organizações e no senso comum.

O espaço comunicacional, tal qual representado e/ou experienciado no mundo globalizado e interconectado, permite a produção de formas diferenciadas de ações coletivas, de expressão de identidades, de conhecimentos, de informações e de culturas. Através das redes sociais é possível indicar mudanças e permanências nos modos de comunicação e transferência de informações. Para Recuero (2009), uma rede é uma metáfora para observar os padrões de conexão de um grupo social, a partir das conexões estabelecidas entre os diversos atores. A abordagem de rede tem seu foco na estrutura social, onde não é possível isolar os atores sociais e nem suas conexões, que são o principal foco do estudo das redes sociais, pois é sua variação que altera as estruturas desses grupos. No contexto de redes sociais surge a interação: atores conectados vão interagir criando a comunicação que é baseada numa relação de ação e reação.

A interação pode ocorrer de forma síncrona (simula uma interação em tempo real) e assíncrona (expectativa de resposta não é imediata). Para Primo (2013), existem duas formas de interação: mútua e reativa. A interação mútua é aquela caracterizada por relações interdependentes e processos de negociação, em que cada interagente participa

da construção inventiva e cooperada da relação, afetando-se mutuamente; já a interação reativa é limitada por relações determinísticas de estímulo e de resposta. Martino (2015) afirma que, em uma rede, as relações entre os participantes dão o tom de seu funcionamento mais do que as características específicas de cada um (caráter relacional). Em outras palavras, trata-se de uma relação entre relações, ou seja, uma perspectiva mútua e recíproca sobre a forma como as pessoas interagem.

Através das redes sociais é possível a formação de grupos, constituídos normalmente por participantes com interesses comuns e/ou afinidades. Estes grupos possibilitam a troca de informações sobre um ou mais assuntos, de forma horizontal (sem hierarquia). Sem cooperação não há grupo, e ela se caracteriza pelo trabalho coletivo em busca de um objetivo comum. Por meio do grupo se dá uma construção coletiva que depende de colaboração entre todos os envolvidos. A competição pode estar diretamente relacionada com uma disputa por recursos, seja no mundo virtual ou no mundo real, podendo ou não gerar conflito. O conflito pode ser altamente benéfico no sentido de fortalecer os grupos, mas também pode desestruturá-los e ocasionar uma ruptura destes. Cabe salientar que o conflito e a competição podem gerar mudanças.

Em um contexto mais amplo, as redes sociais virtuais podem ser classificadas como um tipo de software social. Chatti *et al* (2007) compreendem software social como ferramentas que aumentam as habilidades sociais e colaborativas das pessoas, facilitando a conexão social e a troca de informações. Para Shirky (2003), softwares sociais são aqueles que suportam a interação em grupo. Smith (2007), por sua vez, entende software social como software que permite às pessoas se conectarem por meio de uma comunicação mediada por computador. Neste cenário, surge o *framework* HoneyComb, uma estrutura de referência para o desenvolvimento de softwares sociais.

3.1. *Framework* HoneyComb

O *framework* HoneyComb (Smith, 2007) é um conjunto de sete elementos, chamados blocos de construção (Figura 1), que representam os aspectos fundamentais de softwares sociais.

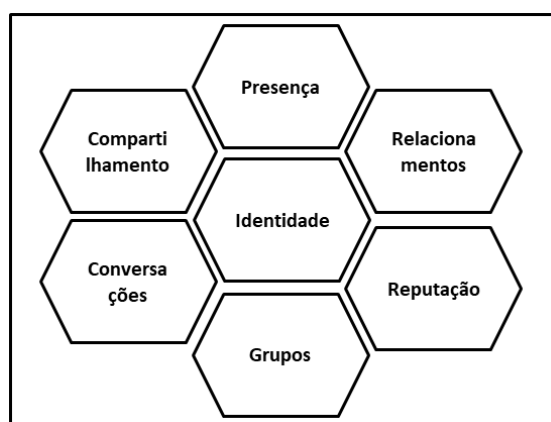


Figura 1. *Framework* HoneyComb

Cada bloco (Tabela 1) representa uma faceta específica de usuários de software social e nem todos eles precisam estar presentes.

Tabela 1. Elementos do *Framework HoneyComb*

Elemento	Descrição
Identidade	elemento central. Refere-se a uma identificação pessoal única dentro do sistema. A expressão de elementos de personalidade de uma pessoa (quem é a pessoal no espaço e no tempo). Ex.: <i>profile</i> do usuário.
Presença	refere-se a algum recurso que permita saber se um usuário está disponível no sistema, compartilhando o mesmo tempo e espaço. Ex.: saber se o usuário está <i>online</i> .
Relacionamentos	compreende uma maneira de representar como os usuários estão conectados entre si. Ex.: amizade (<i>facebook</i>), seguidores (<i>twitter</i>).
Reputação	algum meio ou conjunto de recursos para se conhecer as pessoas no sistema. Ex.: para saber se a pessoa produz informações confiáveis.
Grupos	recursos que promovem a formação de comunidades que compartilham interesses, preferências, ideias, opiniões.
Conversações	recursos pelos quais as pessoas podem se comunicar no sistema. Ex.: mensagens instantâneas, e-mails, fóruns.
Compartilhamento	funcionalidades que permitem que os atores compartilhem fontes de informação sobre o sistema, como documentos, fotos, áudios e vídeos.

Uma análise do *framework HoneyComb* sob a perspectiva da ER foi realizada por Kosvoski e De Bortoli (2018). O comparativo deu-se a partir da atribuição de um grau de importância (muito importante, importante, pouco importante e não se aplica) para a relação dos sete elementos do *framework* a dez características da ER. Na conclusão, foi verificada uma aderência de 78% dos elementos do *framework* com as características da ER consideradas, sendo 55% muito importantes, 18% importantes, 4% pouco importantes e 21% não se aplicaram. A Tabela 2 apresenta um resumo do comparativo realizado.

Tabela 2. Relação das características da ER e do *Framework HoneyComb*

		HONEYCOMB FRAMEWORK						
		Identidade	Presença	Relacionamento	Reputação	Grupos	Conversação	Compartilhamento
CARACTERÍSTICAS ER	Trabalho coletivo	MI	I	NA	MI	MI	MI	MI
	Negociação	MI	MI	NA	MI	MI	MI	MI
	Colaboração	MI	PI	NA	MI	MI	MI	MI
	Comunicação	I	I	NA	MI	MI	MI	MI
	Registro	NA	NA	NA	NA	MI	I	MI
	Diferença cultural	MI	NA	NA	MI	MI	MI	MI
	Geração de ideias	I	NA	NA	MI	MI	MI	MI
	Conflito	I	I	NA	PI	PI	MI	I
	Compartilhamento	I	PI	NA	MI	MI	MI	MI
	Moderação	I	I	NA	MI	MI	I	MI

Os resultados mostram que os elementos do *framework* (exceto o Relacionamento) têm forte relação com a ER, ratificando o caráter social da área. No entanto, para apresentar maior adequação à ER, alguns elementos necessitam ser acrescentados, e outros modificados, o que justifica a proposta de adequação.

4. HoneyComb-ER

Considerando os aspectos sociais da ER, indicados anteriormente, a ascensão das redes sociais e visando o desenvolvimento de uma ferramenta de apoio à ER com base em conceitos de redes sociais, é apresentado o *framework* HoneyComb-ER. O objetivo é disponibilizar um *framework* conceitual baseado no HoneyComb, com elementos que abrangem características específicas da ER, servindo como base para o desenvolvimento de softwares sociais que apoiem essa área.

A metodologia pautou-se em análise bibliográfica e comparativa a partir dos principais autores da área, utilizando dois especialistas com experiência em docência e profissional na área de requisitos. Os especialistas, pesquisadores responsáveis pelo projeto, analisaram o HoneyComb e as características da ER buscando compatibilidade e necessidade de adaptações. Um terceiro especialista foi acionado para mediar conflitos de interpretação. A Figura 2 apresenta o HoneyComb-ER, contendo nove elementos (Tabela 3), onde cada bloco representa uma faceta específica da ER.

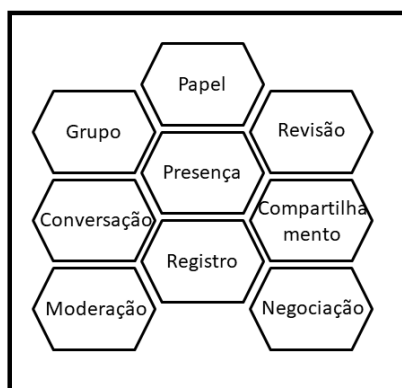


Figura 2. Framework HoneyComb-ER

Optou-se por utilizar nomes no singular e foram mantidos os elementos Presença, Grupo, Conversação e Compartilhamento. Para uma melhor adequação à ER, alguns ajustes foram feitos nas definições e é utilizado o termo Ator para representar os stakeholders.

Tabela 3. Elementos do Framework HoneyComb-ER

Elemento	Descrição
Papel	refere-se a uma identificação do ator no sistema. A expressão do papel desempenhado, sua bagagem cultural e experiência profissional (quem é o ator no espaço e no tempo). O objetivo é poder conhecer a procedência da informação fornecida pelo ator. Ex.: Fernando, contador, graduado em ciências contábeis, especialista em finanças, atua neste papel há cinco anos. Atuou por dois anos como analista financeiro.

Elemento	Descrição
Presença	refere-se a algum recurso que permita saber se um ator está disponível no sistema, compartilhando o mesmo tempo e espaço. A ideia é verificar a possibilidade de comunicação entre atores - tirar uma dúvida, resolver uma pendência, compartilhar um documento ou simplesmente dialogar - no momento que surgir a necessidade.
Grupo	recursos que promovem a possibilidade de formação de equipes de atores para o desenvolvimento de um sistema. Ex.: grupo de atores para definir requisitos de um sistema de contabilidade.
Conversação	recursos pelos quais as pessoas podem se comunicar no sistema. Ex.: mensagens instantâneas, e-mails, fóruns.
Compartilhamento	refere-se a funcionalidades que permitem que os participantes compartilhem algo, como documentos, áudios, fotos e vídeos.
Registro	relaciona-se a funcionalidades que possibilitem o registro dos requisitos definidos pelos atores.
Moderação	prover recursos para conduzir situações de conflito entre os atores e facilitar a decisão, visando o consenso.
Negociação	com estreita ligação ao elemento Moderação, o objetivo é proporcionar mecanismos para obter soluções e acomodar pontos de vista conflitantes dos atores, visando o senso comum.
Revisão	refere-se a funções que permitam a revisão do documento de requisitos elaborado pelos atores, a fim de analisar se está completo, correto e não ambíguo, bem como outros critérios definidos pelo grupo.

Decidiu-se unir Identidade e Reputação em um elemento só: Papel. Julgou-se que os atores desempenham papéis no universo de informações, não sendo consideradas relevantes as informações pessoais no contexto da ER, e sim, o papel que desempenham, bem como, a bagagem cultural adquirida em sua trajetória profissional. O elemento Relacionamento foi suprimido, uma vez que os atores estão conectados através de uma única relação, a parceria para definir requisitos. O elemento Registro foi criado devido à necessidade de formalização das definições em documento de requisitos. Cabe salientar que a definição do formato do documento ficará a cargo dos atores.

É bastante provável que, em algum momento, haja conflito entre os atores no processo de definição de requisitos. Neste contexto, entende-se por conflito a falta de entendimento entre as partes, causado por diferentes pontos de vista sobre algum requisito do software que está sendo desenvolvido. Em situações de conflito, será preciso acionar um processo de comunicação para alinhar os interesses conflitantes, utilizando Moderação. O objetivo não é que algum ator leve vantagem sobre os outros, e sim, chegar a um consenso, evitar excessos e manter o foco. O consenso, por sua vez, consiste em uma ação que permita que os atores cheguem a um acordo que satisfaça a todos, visando sempre atender da melhor forma possível os interesses do

ambiente para o qual o software está sendo construído. A fim de atender o exposto, criou-se o elemento Moderação.

Para facilitar o processo decisório na moderação, será preciso negociar. Assim, surge o oitavo elemento, a Negociação, que pode ser definida como sendo a prática de resolver diferenças, chegar a acordos e encontrar um denominador comum que satisfaça pontos de vista diferentes, neste processo que é decisório e colaborativo. Solucionar conflitos, alinhar interesses conflitantes e obter resultados adequados para as discussões são os principais objetivos da negociação. Por representar uma estrutura social, na ER é preciso lidar com as diferenças culturais (cada ator é especialista em sua área), sem ferir uma das premissas do *framework* que é de não ter hierarquia. Todos os atores devem ter a chance de influenciar, persuadir ou alterar as ideias do outro. É uma chance para o diálogo e é importante assegurar que todos os atores envolvidos na negociação tenham sido ouvidos, mantendo um espaço aberto, com clima amigável, para que todos estejam dispostos a “ouvir” os interesses de cada um, gerando ideias criativas para a satisfação integral do grupo.

Por fim, o elemento Revisão foi criado com o intuito de possibilitar a validação dos requisitos definidos. Assim como no HoneyComb, cada bloco representa uma faceta específica e nem todos precisam estar presentes nas propostas de implementação que terão o HoneyComb-ER como base.

5. Considerações Finais

Por ter forte envolvimento de pessoas, além dos aspectos técnicos, existem aspectos sociais a serem considerados na ER. Surge, assim, uma possibilidade de apoio à ER no contexto de softwares/redes sociais, que favorece a interação de forma virtual, permitindo a transposição contínua de obstáculos entre o mundo físico e o mundo virtual, em um nível de complementaridade entre as interações *online* e àquelas envolvidas *offline*.

Desta forma, despontou o objetivo de criar um *framework* de caráter conceitual que atenda ao mesmo tempo os elementos essenciais de um software social, já presentes no HoneyComb, e às características da ER. O *framework* HoneyComb-ER permite que softwares sociais possam ser criados para apoiar a ER, atendendo as necessidades de ambas as áreas e oportunizando maior aderência do que o HoneyComb. Uma vantagem desta forma de interação é a possibilidade de expressar ideias sem estar frente a frente com outros atores, o que pode encorajar a manifestação de opiniões e de posicionamentos, através de relacionamentos horizontais, sem hierarquia, incentivando cooperação, negociação, gerenciamento de conflitos, compartilhamento e registro, fatores importantes para o sucesso de qualquer projeto de software.

Na prática, requisitos de um sistema são negociados e não descobertos, o desafio está em criar mecanismos para facilitar essa negociação, através de um ambiente amigável, produtivo e organizado. Uma abordagem baseada em rede tem, assim, seu foco na estrutura social, onde não é possível isolar os atores sociais nem sequer suas conexões. No entanto, o HoneyComb-ER é um modelo conceitual e portanto não se configura como uma solução completa para todos os desafios da ER, podendo ser integrada com outras abordagens. Além disso, projetos mais robustos podem exigir uma abordagem mais minuciosa e estratégias adicionais para gerenciar eficientemente os recursos. Como trabalho futuro, pretende-se elaborar diretrizes

detalhadas do uso do *framework*, bem como orientações aos profissionais sobre sua integração com as demais etapas da ER. Também almeja-se desenvolver um protótipo de ferramenta para apoiar a ER, com base no *framework* HoneyComb-ER. Finalizando, salienta-se que este estudo foi financiado pelo IFRS.

5. Bibliografia

- Boehm, B. e Egyed, A. (1998). Software Requirements Negotiation: Some Lessons Learned, Proc. Intl. Conf. Software Engineering, ACM/IEEE. p.503-06.
- Chatti M. A., Jarke, M., Wilke D. F. (2007). The future of e-learning: a shift to knowledge networking and social software. In: Journal of Knowledge and Learning, v.3, n. 4-5, p. 404-420.
- Hooshyar, H. et al. (2023). Impact in Software Engineering Activities After One Year of COVID-19 Restrictions for Startups and Established Companies. IEEE Access, v.11, p. 55178-55203.
- Jirotko, M. e Goguen, J. (1994). Requirements Engineering - social and technical issues. Academic Press.
- Kosvoski, D e De Bortoli, L. Â. (2018). Análise do framework honeycomb sob a perspectiva da engenharia de requisitos In: II Escola Regional de Engenharia de Software. Anais da 2a Escola Regional de Engenharia de Software. p.1-8.
- Leite, J. C. P. e Doorn, J. H. (2003). Perspectives on Software Requirements. Springer.
- Marteleto, R. M. (2010). Redes Sociais, mediação e apropriação de informações: situando campos, objetos e conceitos na pesquisa em Ciência da Informação. Pesq. bras. ci. inf., Brasília, v.3, n.1,p.27-46, jan./dez.
- Martino, L. M. S. (2015). Teoria das Mídias Digitais. Vozes.
- Pressman, R. (2016). Engenharia de Software - uma abordagem profissional. McGraw-Hill.
- Primo, A. (2013). Interações em rede. Sulina.
- Recuero, R. (2009). Redes sociais na internet. Sulina.
- Shirky, C. (2003). A group is its own worst enemy: social structure and social software. In: O'Really Emergen Technology Conference, Santa Clara.
- Smith, G. (2007) "Social Software Building Blocks", <http://nform.com/publications/social-software-building-block>.
- Sommerville, I. (2011). Engenharia de Software. Addison Wesley.
- Sommerville, I. e Sawyer, P. (1997). Requirements Engineering - A good practice guide. John Wiley & Sons.
- Yu. E. S. (1997). Towards Modelling And Reasoning Support For Early-Phase Requirements Engineering. 3rd IEEE Int. Symp. on Requirements Eng., 226-235.
- Yu, E. S. (2009). Social Modeling and i*. In: A. Borgida, V. Chaudhri, P. Giordini, E. Yu (eds.). Conceptual Modeling: Foundations and Applications - Essays in Honor of John Mylopoulos. LNC vol. 5600, Springer.