



UNIVERSIDADE FEDERAL DO CEARÁ
CAMPUS DE RUSSAS
CURSO DE ENGENHARIA DE SOFTWARE

UBIRATAN TEIXEIRA DE CARVALHO

**UTILIZANDO O FRAMEWORK SRAAF (SOFTWARE REQUIREMENT
AMBIGUITY AVOIDANCE FRAMEWORK) NO DESENVOLVIMENTO
DISTRIBUÍDO DE SOFTWARE: UM ESTUDO DE CASO**

RUSSAS

2022

UBIRATAN TEIXEIRA DE CARVALHO

UTILIZANDO O FRAMEWORK SRAAF (SOFTWARE REQUIREMENT
AMBIGUITY AVOIDANCE FRAMEWORK) NO DESENVOLVIMENTO
DISTRIBUÍDO DE SOFTWARE:
UM ESTUDO DE CASO

Trabalho de conclusão de curso apresentado ao Curso de Graduação em Engenharia de Software da Universidade Federal do Ceará Campus Russas, como requisito parcial à obtenção do grau de Bacharel em Engenharia de Software.

Orientadora: Profa. Dra. Patrícia Freitas Campos de Vasconcelos.

RUSSAS

2022

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas

Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

C329u Carvalho, Ubiratan Teixeira de.

Utilizando o framework Sraaf (Software Requirement Ambiguity Avoidance Framework) no desenvolvimento distribuído de software: um estudo de caso / Ubiratan Teixeira de Carvalho. – 2022. 53 f. : il. color.

Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Russas, Curso de Engenharia de Software, Russas, 2022.

Orientação: Profª. Dra. Patrícia Freitas Campos de Vasconcelos.

1. SRAAF. 2. DDS. 3. elicitação. 4. técnicas. 5. engenharia de requisitos. I. Título.

CDD 005.1

UBIRATAN TEIXEIRA DE CARVALHO

UTILIZANDO O FRAMEWORK SRAAF (SOFTWARE REQUIREMENT
AMBIGUITY AVOIDANCE FRAMEWORK) NO DESENVOLVIMENTO
DISTRIBUÍDO DE SOFTWARE: UM ESTUDO DE CASO

Trabalho de conclusão de curso
apresentado ao Curso de Graduação em
Engenharia de Software da Universidade
Federal do Ceará Campus Russas, como
requisito parcial à obtenção do grau de
Bacharel em Engenharia de Software.

Aprovada em 18/12/2022

BANCA EXAMINADORA

Profª. Dra. Patrícia Freitas Campos de Vasconcelos (Orientador)
Universidade Federal do Ceará (UFC)

Profa. Dra. Jacilane de Holanda Rabelo
Universidade Federal do Ceará (UFC)

Prof. Ms. José Osvaldo Mesquita Chaves
Universidade Federal do Ceará (UFC)

AGRADECIMENTOS

Quero agradecer primeiramente a Deus pela oportunidade de permitir que este momento se tornasse realidade e por me fortalecer na perseverança, fazendo eu acreditar que seria possível chegar ao fim da minha jornada de curso com sucesso.

Agradecer também a minha família, principalmente aos meus pais, minha irmã Vanessa, por toda paciência e compreensão, e que me apoiou e incentivou a nunca desistir. Também agradeço a todas as pessoas que eu conheci nesse tempo de graduação ao longo dos anos, que contribuíram de forma valorosa para o meu conhecimento e evolução nesta jornada.

Também quero agradecer à minha namorada Joyce por todo carinho, respeito e por tornar todos os dias da minha vida mais felizes e por todo o apoio que ela me deu nos momentos mais difíceis.

Obrigado aos amigos que ganhei na faculdade, Lucas, Tiago, Letícia, Luiz, Guilherme, Matheus, Lauana pelos momentos difíceis e felizes que me acompanharam nesta jornada. Finalmente, gostaria de expressar minha gratidão à professora Patrícia por sua orientação, apoio e por acreditar em mim. Com toda a sua experiência trabalhando para mim, ela possibilitou que eu concluísse esta tarefa e garantiu que eu seguisse o caminho certo para alcançar meu objetivo final.

Por fim, gostaria de expressar minha eterna gratidão a todos que contribuíram direta ou indiretamente para a conclusão deste trabalho.

*“Para os dias bons: sorrisos.
Para os dias ruins: paciência.
Para todos os dias: fé.”*

RESUMO

Com o passar dos anos a humanidade sempre nos surpreende com a construção de novas tecnologias em diferentes contextos que provocam grande impacto na sociedade para uma melhor qualidade de vida, porém as empresas de tecnologia responsáveis pela sua construção enfrentam vários problemas que perduram até hoje durante o processo de desenvolvimento de um software. Um desses problemas envolve a etapa de elicitação de requisitos que representa um papel crítico no desenvolvimento do sistema, o que acaba necessitando de uma maior atenção por parte dos engenheiros e analistas devido a ambiguidade de requisitos de software que possam vir a surgir. Além de tudo, com o surgimento do vírus SARSCoV no início de 2020 fez com que empresas de tecnologia espalhadas pelo mundo se adequassem ao DDS (Desenvolvimento Distribuído de Software) mudando a sua realidade de negócio levando em consideração seus aspectos culturais, organizacionais e sociais. O framework Software Requirement Ambiguity Avoidance Framework (SRAAF), auxilia na escrita de requisitos sem ambiguidade, ajudando na seleção da técnica de elicitação correta visando uma melhoria no que chamamos de SRS (Software Requirements Specification). Este trabalho visa a aplicação de um estudo de caso utilizando o framework SRAAF em equipes que trabalham de forma remota, utilizando como estudo uma empresa do ramo de *E-commerce* que atualmente trabalha com a modalidade *home-office* devido a pandemia da Covid-19, visando trazer resultados obtidos com a aplicação do framework SRAAF na escrita de requisitos.

Palavras-chave: SRAAF; requisitos de software; elicitação; técnicas; levantamento requisitos; engenharia de requisitos; DDS; home-office.

ABSTRACT

Over the years, humanity has always surprised us with the construction of new technologies in different contexts that have a great impact on society for a better quality of life, but technology companies are responsible for their construction, facing various problems that last until today during the software development process. One of these problems involves a requirements elicitation step that plays a critical role in system development, which ends up requiring greater attention from engineers and analysts due to the ambiguity of software requirements that may arise. In addition, the SARSCoV virus in early 2020 caused technology companies around the world to adapt to DDS (Distributed Software Development), changing their business reality, taking into account their cultural, organizational and social aspects. The framework Software Requirement Ambiguity Avoidance Framework (SRAAF) assists in writing requirements without ambiguity, helping in the selection of the correct elicitation technique aiming at an improvement in the SRS (Software Requirements Specification) provided. This work aims to apply a case study using the SRAAF framework in teams that work remotely, using as a study an E-commerce company that currently works with the home-office modality due to the Covid- 19 pandemic, seeking to bring results obtained with the application of the SRAAF framework in the writing of requirements.

Keywords: SRAAF; software requirements; elicitation; techniques; requirements survey; requirements engineering; DDS; home-office.

LISTA DE FIGURAS

Figura 1 - Etapas da Engenharia de Requisitos.....	13
Figura 2 - Processo do JAD.....	17
Figura 3 - Representação diagrama de casos de uso.....	19
Figura 4 - Modelo de Prototipagem.....	20
Figura 5 - Etapas básicas do framework SRAAF.....	22
Figura 6 - Parâmetros significativos do projeto.....	23
Figura 7 - Valores dos atributos significativos(engenheiro de requisitos).....	24
Figura 8 - Procedimentos metodológicos.....	32
Figura 9 - Etapas do estudo de caso.....	35
Figura 10 - Processo do SRAAF.....	37
Figura 11 - Experiência com framework SRAAF.....	39
Figura 12 - Rede de sugestões de melhoria dos parâmetros significativos.....	40
Figura 13 - Rede de técnicas de elicitação listadas no framework.....	42
Figura 14 - Rede de dificuldades na aplicação da técnica de elicitação.....	43
Figura 15 - Rede de características do framework que auxilia na etapa da ER.....	44
Figura 16 - Rede de Desafios/dificuldades da utilização do framework SRAA.....	46

LISTA DE TABELAS

Tabela 1 - Trabalhos Relacionados.....	31
Tabela 2 - Descrição das Etapas para execução do SRAAF.....	34
Tabela 3 - Sugestões de melhoria dos parâmetros significativos.....	41
Tabela 4 - Técnicas de elicitação listadas no framework.....	43
Tabela 5 - Dificuldades na aplicação da técnica de elicitação.....	44
Tabela 6 - Características do SRAAF auxilia na etapa da ER.....	45
Tabela 7 - Desafios/dificuldades da utilização do framework.....	46

LISTA DE ABREVIATURAS E SIGLAS

ER Engenharia de Requisitos

JAD *Joint Application Development*

UML Unified Modeling Language

DDS Desenvolvimento Distribuído de Software SRS *Software Requirements Specification*

SRAAF Software Requirement Ambiguity Avoidance Framework

GT Grounded theory

SUMÁRIO

1	INTRODUÇÃO.....	12
1.1	OBJETIVO GERAL.....	15
1.1.1	OBJETIVOS ESPECÍFICOS.....	15
2	FUNDAMENTAÇÃO TEÓRICA.....	
3	TRABALHOS RELACIONADOS.....	30
4	PROCEDIMENTOS METODOLÓGICOS.....	36
5	ESTUDO DE CASO.....	37
5.1	PLANEJAMENTO DA APLICAÇÃO DO PROCESSO NO AMBIENTE REMOTO.....	38
5.2	CONDUÇÃO DO ESTUDO DE CASO.....	39
5.3	ANÁLISE DOS RESULTADOS OBTIDOS.....	40
5.3.1	ANÁLISE QUANTITATIVA DO USO DO FRAMEWORK.....	40
5.3.2	ANÁLISE QUALITATIVA DO USO DO FRAMEWORK.....	42
6	MELHORIA DO PROCESSO DO SRAAF.....	51
7	CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS.....	51
	REFERÊNCIAS.....	53
	APÊNDICE A- QUESTIONÁRIO UTILIZADO PARA AVALIAR O ESTUDO DE CASO NO PROJETO.....	57
	APÊNDICE B- PERGUNTAS QUALITATIVAS PARA AVALIAÇÃO DA APLICAÇÃO DO FRAMEWORK.....	58

1 INTRODUÇÃO

A tecnologia em boa parte da sua história vem sendo aprimorada cada vez mais com a inclusão de novos avanços e técnicas, que buscam suprir as necessidades da sociedade em diferentes contextos e âmbitos da vida moderna. Sendo essa demanda de serviços e dispositivos, desenvolvidos por pequenas e grandes empresas de software espalhadas por todo o mundo. Uma pesquisa feita pela Associação Brasileira de Software (ABES) mostra que a indústria da tecnologia cresceu cerca de 22,9% e investiu por volta de US\$ 49,5 bilhões, ocupando a décima posição no ranking mundial somente no ano de 2021. E mesmo com todo esse avanço recente, o desenvolvimento de software em sua evolução com os anos, sempre veio acompanhada de certas dificuldades inerentes a mão de obra especializada escassa para produção de software.

Dessa forma o DDS (desenvolvimento distribuído de software) foi se tornando cada vez mais empregado nas empresas ao redor do mundo. Tal forma de desenvolvimento de software se transformou em um aspecto primordial com o cenário Pandêmico do Covid-19 em 2020, obrigando de forma repentina a adequação de muitas empresas de tecnologia a assumir o home.

O DDS em um contexto de afastamento social obrigatório se tornou fundamental para que diversos setores da sociedade conseguissem se adequar ao cenário mundial da pandemia. Os ofícios dos mais variados âmbitos sociais tiveram que se transformar digitalmente para conseguir se sustentar na pandemia, o que acelerou mais ainda a demanda por desenvolvimento de software por parte das empresas. Essa aceleração mostrou a necessidade de buscar e alocar profissionais qualificados de diferentes locais para trabalhar em prol de projetos de qualquer lugar do país e até do mundo.

De acordo com Carmel (1999), o DDS se configura como um desenvolvimento de software onde as equipes do projeto estão espalhadas de forma geográfica. O motivo da propensão em relação ao desenvolvimento distribuído de software leva em consideração a necessidade das empresas de software em diminuir os custos de desenvolvimento, mão de obra especializada, e a necessidade de estarem mais próximas a partes interessadas do projeto.

Entretanto, ainda que o DDS possua suas vantagens, ela também tem suas ressalvas que segundo Arttu (2008) a comunicação é o problema mais comum em projetos de empresas que adotam o DDS, já que as relações sociais entre os membros se tornam desafiadoras, devido aos fatores humanos e a limitação dessa comunicação, além de outros problemas como a distância e fatores culturais. Ainda assim, esse fato influencia diretamente na etapa da

engenharia de requisitos devido a dificuldade de comunicação para extrair requisitos assertivos e sem ambiguidades que possam vir a surgir durante o processo de elicitação dos requisitos, etapa que desempenha um papel fundamental no processo de desenvolvimento de um software de sucesso. Porém, apesar do crescente aumento das empresas de software, as mesmas vêm enfrentando problemas durante todo o seu processo de desenvolvimento em termos de controle de qualidade, principalmente na fase de Engenharia de Requisitos (ER), que pode ser vista como uma das atividades mais críticas para o processo de desenvolvimento de software, visto que o mesmo se tornou cada vez mais exigente quando falamos de qualidade do produto e diferencial perante seus concorrentes.

É nesse ponto que entra a etapa da Engenharia de Requisitos que tem como um dos objetivos definir requisitos que representem com o máximo de precisão possível as necessidades das partes interessadas (DEEPTI, 2008).

A Elicitação de Requisitos é uma das atividades mais cruciais no processo da ER. A mesma consiste na etapa inicial do processo de engenharia de requisitos, e visa reunir todos os requisitos relevantes por meio de interações diretas ou indiretas entre analistas de requisitos e as partes interessadas. Pois, como um sistema se faz por meio de requisitos que é por onde se fomenta todas as funcionalidades de um sistema, isso pode acabar tornando impossível sua descoberta sem uma técnica de elicitação.

Segundo Standish Group (2020), organização que reporta periodicamente estatísticas sobre sucesso e fracasso de projetos de software, no ano de 2020, 19% dos projetos de software iniciados fracassaram, 50% foram entregues com falhas detectadas, atrasados ou com o orçamento estourado e somente 31% foram finalizados com sucesso. Se comparado ao ano de 2017 a taxa de sucesso foi avaliada em 36%. Nesse contexto, podemos afirmar que produzir um software 100% funcional não é uma tarefa fácil, devido à dificuldade de se entender as demandas solicitadas pelas partes interessadas.

Contudo, segundo Paola (2017) mesmo que o problema de elicitação de requisitos não seja novo e tenha sido abordado várias vezes ao longo dos anos, ainda é considerado um dos mais desafiadores do processo de engenharia de requisitos. A maior parte do sistema falha devido ao uso de práticas de elicitação incorretas, que acaba levando ao surgimento de ambiguidades, produzindo assim um SRS (Software requirements specification) de má qualidade, ou também ao surgimento de novos requisitos após sua elicitação, gerando um retrabalho dos requisitos, e uma reelaboração de sua especificação, aumentando o custo para correção desses erros. Esses aspectos podem ser piorados se for considerado o contexto do

DDS, já que essa etapa envolve uma maior comunicação entre as partes interessadas.

Com base nas informações ditas a ambiguidade é um problema muito corriqueiro na fase da elicitação de requisitos, já que os requisitos elicitados podem ser descritos utilizando-se da linguagem natural. Tal fato pode tornar a escrita dos requisitos ambígua e até levar a interpretações equivocadas, com pouca clareza na escrita desta documentação. Apesar de existirem técnicas que ajudam a amenizar a ambiguidade, ainda assim isto não é suficiente para lidar com esse impasse.

Todavia, tentar amenizar as ambiguidades em etapas iniciais do projeto é de fundamental importância, uma vez que ajuda a amenizar os riscos que possam surgir em etapas posteriores do projeto, pois se encontrado no final do projeto é mais caro do que se fosse encontrado no início.

No mercado atual é possível encontrar diversas técnicas de elicitação de requisitos, onde pode-se citar algumas delas como entrevista, brainstorming, etnografia e histórias de usuário. Devido ao surgimento de novas técnicas se faz necessário uma escolha correta da técnica visando identificar as necessidades relevantes do projeto, e reduzir as divergências entre as partes interessadas com o propósito de filtrar os requisitos para atender às restrições das partes interessadas.

Entretanto, os analistas de negócios possuem dificuldades em relação a qual técnica escolher, e determinar se aquela técnica realmente irá atender as necessidades do produto final. Selecionar a técnica de elicitação de requisitos apropriada de acordo com as características do projeto é importante para a conclusão bem-sucedida da fase da Engenharia de Requisitos (DEEPTI, 2008).

É nesse ponto que entra o SRAAF (*Software Requirement Ambiguity Avoidance Framework*)¹. O SRAAF é um framework que envolve a etapa da elicitação de requisitos, na qual tem como proposta evitar a ambiguidade nos requisitos na fase inicial do desenvolvimento do documento de requisitos e ajudar a selecionar a técnica de elicitação de requisitos mais efetiva com base em alguns parâmetros. Para tanto pretende-se aplicar o framework SRAAF [Kumar (2019)] em um projeto de e-commerce em uma empresa de desenvolvimento de software. Por questões de privacidade a empresa será denominada de forma fictícia como "EmpresaV". Como dito anteriormente, esse framework objetiva ajudar na escolha da melhor técnica de elicitação de requisitos a ser aplicada, na construção do documento de requisitos com menor ambiguidade e melhor qualidade. Além disso,

1 Maiores detalhes do framework serão apresentados na seção 2.4.

pretende-se verificar se o framework SRAAF utilizado neste estudo de caso foi eficiente, levando-se em consideração o contexto *home-office*.

1.1 Objetivo geral

Investigar a utilização do framework SRAAF desenvolvido por Ashok Kumar (2019) no contexto do desenvolvimento distribuído de software em um estudo de caso.

1.1.1 Objetivos específicos

- Aplicar o framework SRAAF no projeto de e-commerce voltado para vendas via whatsapp;
- Analisar a eficácia do framework SRAAF no auxílio à construção do documento de especificação de requisitos de software(SRS);
- Oferecer informações sobre o uso da técnica SRAAF na etapa da Engenharia de requisitos;
- Levantar os principais problemas e dificuldades em relação a aplicação do framework SRAAF.

2 FUNDAMENTAÇÃO TEÓRICA

Nesta seção serão apresentados os conceitos utilizados para um melhor entendimento desta pesquisa.

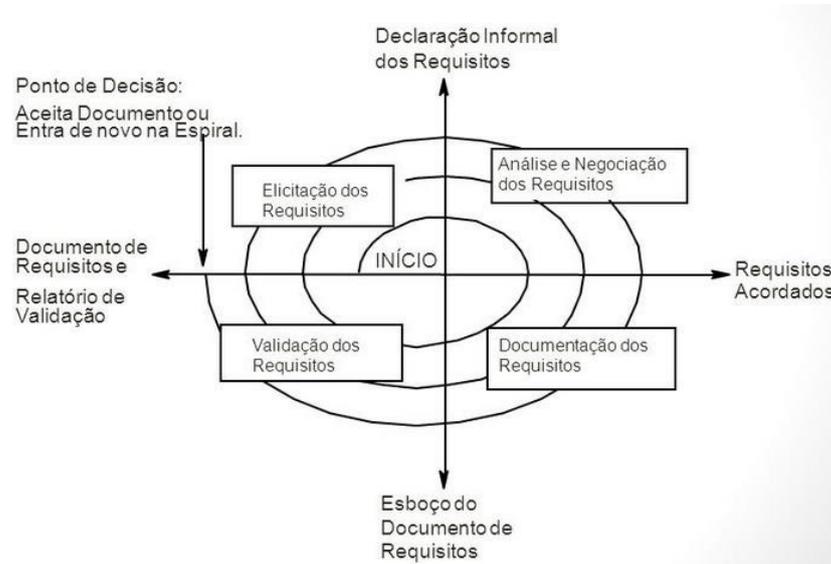
2.1 Engenharia de Requisitos

Em qualquer projeto independente do setor a que se destine, é necessária a elaboração de requisitos, já que esses devem atender às necessidades para qual aquele projeto foi destinado. Isso não é muito diferente quando falamos de desenvolvimento de software, já que o sistema deve atender a todas as partes interessadas, onde esses requisitos devem possuir clareza de acordo com o sistema que foi proposto. Diante disso, podemos afirmar que os requisitos são utilizados como base para o desenvolvimento do projeto.

De acordo com Sampada (2020) uma das principais suspeitas para o colapso do sistema de software é a má especificação dos requisitos. Sendo assim a construção dos requisitos de um software primeiramente se dá pela etapa da engenharia de requisitos.

Segundo Sommerville (2007), a engenharia de requisitos tem como objetivo definir o que o sistema deve fazer, quais as necessidades reais e identificar quais restrições existem para que o software seja desenvolvido. A engenharia de requisitos é nada mais do que um processo que envolve a padronização do gerenciamento do projeto, visando a garantia e sucesso do projeto com uma entrega eficiente e de qualidade em alinhamento com as partes interessadas, dentro do prazo estipulado. A ER possui 4 atividades básicas que visam garantir o sucesso do projeto (ver Figura 1).

Figura 1 - Etapas da Engenharia de Requisitos



Fonte: Falbo (2012).

A primeira atividade se dá pela elicitação de requisitos. É nessa etapa que são levantadas as necessidades das partes interessadas do software, ou seja, é o processo de adquirir os requisitos do sistema, como também suas restrições. O objetivo principal dessa etapa é atender os requisitos que foram levantados de forma correta para atender as necessidades das partes interessadas visando obter o requisito correto.

Nessa etapa tem-se a necessidade de 4 conhecimentos básicos por parte da equipe, são eles:

- Entendimento do domínio na qual o software será aplicado;
- Entendimento do problema, ou seja, o que aquele software irá resolver;
- Entendimento do negócio, como aquele software irá impactar na organização;
- Entendimento das necessidades das partes interessadas, que envolve os processos e demandas para realizar o trabalho.

Porém existem diversas técnicas que são responsáveis por realizar a coleta desses requisitos, tais como entrevista, questionário, observação, *Joint Application Development* (JAD), casos de uso, prototipação.

Nesse contexto, vale ressaltar que apesar das diversas técnicas de elicitação existentes, onde algumas são mais utilizadas que outras, ainda assim existem muitas dificuldades por parte dos profissionais em identificar qual técnica escolher. Onde segundo Dante (2016)

existe uma falta de métodos e ferramentas para selecionar a técnica mais adequada a ser usada a qualquer momento em um projeto de desenvolvimento de software.

2.2 Técnicas de Elicitação

Técnica de Elicitação pode ser definida como a maneira de fazer algo ou método prático aplicado a alguma tarefa específica. As técnicas de elicitação são ferramentas de localização e compreensão exata para o bem das partes interessadas (Nikita, 2015).

O objetivo da técnica de Elicitação é descobrir o máximo de problemas possíveis, para que seja mais fácil para as partes interessadas obter a aplicação mais adequada de acordo com os requisitos (Nikita, 2015). Hoje existem várias técnicas de elicitação de requisitos, as principais dentre elas são:

- Entrevista;
- Questionário;
- Observação;
- *Joint Application Development (JAD)*;
- Casos de uso;
- Prototipação;

2.2.1 Entrevista

A técnica de entrevista de acordo com Brum (2011) é uma das técnicas tradicionais mais simples de se utilizar pelo fato de a mesma produzir bons resultados na fase inicial do projeto para obtenção de informações. Segundo Sommerville (2007), a entrevista constitui-se de 2 tipos que seriam a entrevista formal e informal.

Na entrevista formal são elaboradas perguntas pré-definidas que são respondidas de forma sequencial evitando que a parte interessada desvie o foco. Já a entrevista informal também possui perguntas pré-definidas, porém para obter maiores informações pode ser necessário uma conversa com o cliente de forma mais descontraída, para que essas informações sejam melhor esclarecidas a fim de serem repassadas ao analista visando uma melhor compreensão.

De acordo com Carvalho (2009) a técnica de entrevista possui 3 etapas, são elas:

- Planejamento;

- Condução;
- Elaboração do relatório.

Na etapa de planejamento é onde são elaboradas as perguntas que irão ser utilizadas na entrevista. Além disso são definidas também quem serão os entrevistados, de preferência os usuários mais relevantes para o projeto. Após a seleção dos entrevistados é marcado uma data e horário para realização da entrevista, onde essa entrevista deverá ser registrada seja com um gravador ou anotações que servirá como documentação. A segunda etapa é a ‘condução’, onde o entrevistador/analista realiza as perguntas aos usuários, e logo em seguida é elaborado um relatório de entrevista contendo todas as informações necessárias levantadas.

2.2.2 Questionário

A técnica de questionário se comparada a entrevista são muito parecidas, já que o questionário possui perguntas que devem ser respondidas pelos usuários, porém a diferença entre elas seria basicamente que o questionário é melhor aplicado quando se deseja atingir um maior número de pessoas. Por exemplo, a técnica de entrevista pode ajudar a refinar respostas obtidas através de perguntas presentes no questionário.

A técnica de questionário é composta de 2 tipos que seriam: objetivo e subjetivo. O questionário objetivo possui respostas claras e objetivas, evitando assim que as respostas obtidas de diferentes usuários estejam distintas. Já o questionário subjetivo possui perguntas abertas que permitem ao usuário descrever sua opinião de forma mais abrangente. A técnica de questionário pode ser aplicada a um grande número de usuários da organização, sendo possível ser respondido pelo cliente de acordo com sua disponibilidade, ou até mesmo remotamente (FABIANO, 2012).

2.2.3 Observação

A etnografia como também é conhecida corresponde a uma técnica de observação que tem como objetivo compreender os requisitos tanto sociais, como organizacionais, além de tentar entender a política organizacional. Nesse caso, o analista se insere no ambiente de trabalho onde o sistema será utilizado para o levantamento de possíveis requisitos. O principal objetivo dessa técnica é identificar requisitos implícitos do sistema que acabam refletindo em processos reais, ou seja, requisitos derivados de como o usuário do sistema realmente realiza o seu trabalho. De acordo com Sommerville (2007, p. 104) o mesmo sugere que o usuário seja observado por uma pessoa imparcial na execução de sua tarefa para que todas as informações necessárias para o desenvolvimento sejam coletadas, sem que o usuário que está sendo observado se sinta intimidado.

2.2.4 Joint Application Development

Segundo Jobs (2010) cita que um dos problemas mais recorrentes no levantamento de requisitos são os diferentes pontos de vista que são apresentados por diferentes pessoas, além de que um requisito pode sofrer alterações no decorrer do projeto. É nesse ponto que entra a técnica JAD.

A técnica JAD é utilizada quando se pretende reunir todas as partes interessadas para levantar os requisitos em um curto prazo de tempo. De acordo com JAD Guidelines (2012) às etapas são bastante estruturadas o que acaba permitindo produzir um documento de qualidade em um curto espaço de tempo, onde a mesma auxilia os desenvolvedores a definirem o escopo do projeto e seus objetivos de forma mais rápida.

Existem 6 tipos de participantes, são eles: líder da sessão, patrocinador, escriba, documentador, especialistas e observadores. O líder da sessão é o responsável por definir os objetivos e planos da sessão, ou seja, ele seria nada mais que um 'facilitador' ajudando na organização de decisões e solucionando conflitos que possam vir a existir. É recomendado que o líder da sessão seja imparcial e que possua experiência com reuniões JAD.

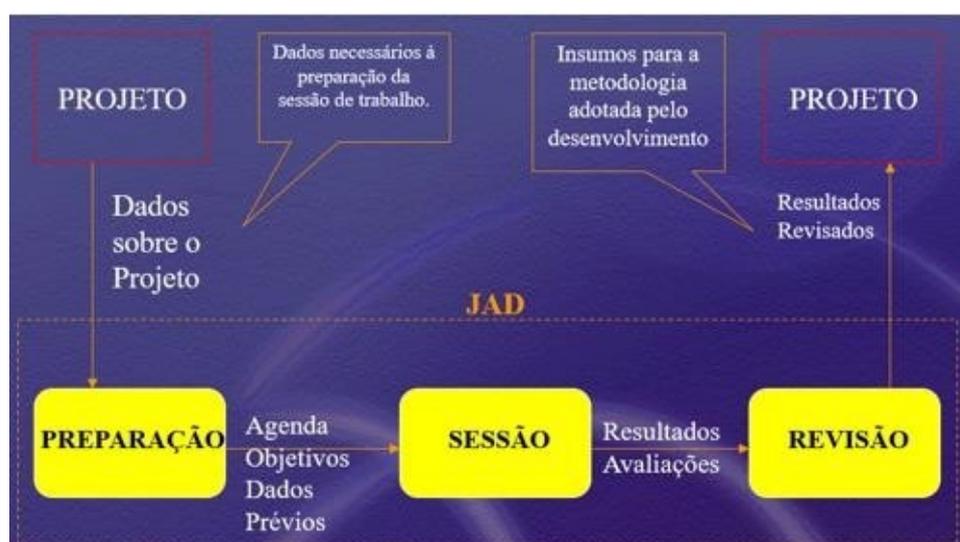
Outro participante seria o 'patrocinador' que é responsável por financiar o projeto a ser desenvolvido. Já o 'escriba' como o próprio nome já diz, tem como objetivo registrar tudo que ocorre nas reuniões podendo ser visualizado por todos os participantes. O participante 'documentador' é o responsável por produzir o documento de requisitos que será utilizado durante todo o processo de desenvolvimento. O 'especialista' simplesmente seria aquele que

possui total conhecimento sobre o processo de negócio. Por último temos o ‘observador’ que seria nada mais que pessoas que não se manifestam durante as reuniões, normalmente esses observadores são representados pela equipe de desenvolvimento.

Levando em conta a função que os participantes exercem, o processo JAD é organizado em 3 etapas conforme Figura 2.

- Preparação;
- Sessão;
- Revisão.

Figura 2 - Processo do JAD



Fonte: David (2012).

A etapa de ‘preparação’ tem como função identificar os participantes, além de definir os produtos e objetivos que serão obtidos e providenciar os recursos que serão necessários. Nessa etapa ocorre a preparação da sessão, onde é realizado o levantamento dos objetivos da sessão, metas do projeto e as regras de conduta que devem ser seguidas nas sessões para se obter uma maior produtividade. Por fim é realizada uma revisão do artefato gerado na sessão.

Já a etapa de ‘sessão’ tem como função verificar se todas as etapas programadas foram abordadas antes mesmo do encerramento da sessão, onde esse trabalho é realizado pelo ‘facilitador’ como já mencionado anteriormente.

Por último, é realizada a etapa de ‘revisão’ onde a documentação passa por uma revisão, porém caso exista algum problema, a mesma será corrigida e repassada aos participantes como um documento final.

Como resultado disso, Yatco (1999) cita que a técnica JAD pode reduzir entre 20% a 50% o tempo para desenvolver o sistema, já que todo o processo de negócio é discutido por todos os membros que estão envolvidos em um curto prazo. Também segundo Jobs (2010) a técnica ajuda a promover o trabalho em equipe, já que todos os membros acabam se envolvendo para buscar a solução mais adequada para resolver determinado problema.

2.2.5 Casos de uso

A técnica de casos de uso tem como objetivo identificar as interações que ocorrem de forma interna no sistema. Onde essas interações podem ser descritas através de cenários que são utilizadas pelos analistas no levantamento de requisitos, ou seja, como as funcionalidades se relacionam umas com as outras e como serão utilizadas pelo usuário.

Esses cenários possuem um fluxo principal para a funcionalidade, porém caso ocorra algum erro o sistema deve tratar esse erro e exibi-lo ao usuário, onde por fim é apresentado o resultado que o sistema deve obter após a execução de determinada tarefa. Portanto, essas informações normalmente são descritas utilizando-se do uso de notações, por exemplo, o UML (Unified Modeling Language) que serve para auxiliar na visualização da funcionalidade em si.

Segundo Pressman (2006, p. 130) devem ser identificados os atores que interagem com alguma funcionalidade do sistema, onde as interações que existem com esses atores devem ser levantadas. Com isso é criado o diagrama de casos de uso conforme Figura 3.

Figura 3 - Representação diagrama de casos de uso



Fonte: Sommerville (2007).

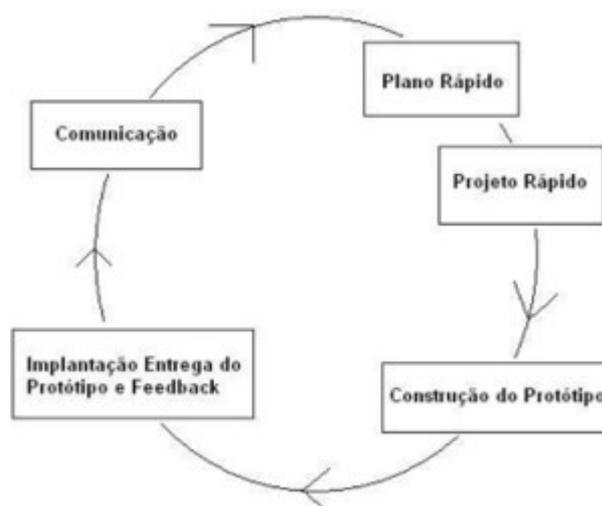
Após essa identificação é criado um documento final contendo todos os casos de uso possíveis na qual representam todas as interações que existem no sistema.

2.2.6 Prototipação

A técnica de prototipação envolve a criação de protótipos contendo uma versão inicial do que pode ser o sistema, pois a mesma ajuda o usuário a definir com clareza quais requisitos são necessários para o desenvolvimento do sistema, já que o usuário não sabe ao certo o que realmente deseja.

Sendo assim, diante do que foi apresentado, um modelo de prototipagem é utilizado com suas respectivas etapas como mostrado na Figura 4.

Figura 4 - Modelo de Prototipagem



Fonte: Domingos (2008).

Na etapa de ‘comunicação’ é onde ocorre a interação entre o usuário e o analista para levantamento prévio de alguns requisitos que irão compor o protótipo. Esse levantamento pode ser feito aplicando-se outras técnicas como entrevistas ou questionários. Na próxima etapa é construído um protótipo rápido com suas respectivas funcionalidades, onde o usuário visualiza o protótipo e verifica se o mesmo está de acordo com o que foi solicitado por ele. Por fim é realizado um ‘feedback’ por parte de quem utilizou o protótipo a fim de relatar as considerações finais. Esse ciclo se repete enquanto os requisitos não satisfizerem a necessidade da parte interessada.

Contudo, existem 2 tipos de protótipo: Protótipo ‘não-operacional’ e Protótipo ‘arranjado às pressas’. O não-operacional é aquele que só é possível visualizar suas telas não sendo possível executá-lo. Já o protótipo arranjado às pressas é possível realizar a sua

execução, ou seja, é permitido visualizar o funcionamento do sistema de forma interativa. No entanto, esse protótipo só é utilizado para que o usuário valide se os requisitos estão de acordo com o que foi solicitado, onde posteriormente será criado o sistema em si com aquilo que foi levantado.

Nesse contexto, é possível visualizar diferentes técnicas que podem vir a ser utilizadas na etapa de elicitaco de requisitos, cada um com suas caractersticas particulares. Neste projeto de pesquisa, abordaremos a eficcia da aplicao do framework SRAAF na escolha das tcnicas existentes para melhoria do processo da ER levando em considerao o aprimoramento da qualidade do documento SRS.

2.3 Software requirements specification (SRS)

Visto a necessidade de termos a etapa da ER como sendo uma das mais essenciais para o desenvolvimento de um software. Essa etapa produz vrios artefatos, porm  produzido um artefato muito importante e essencial para o sucesso do software, esse artefato  conhecido como SRS. O SRS  um Documento de Especificao de Requisitos de Software que descreve o sistema de software a ser desenvolvido, ele contm tudo que  necessrio para se desenvolver o sistema, como requisitos do sistema, regras de negcios, interfaces, casos de uso, etc (LIMA, 2022).

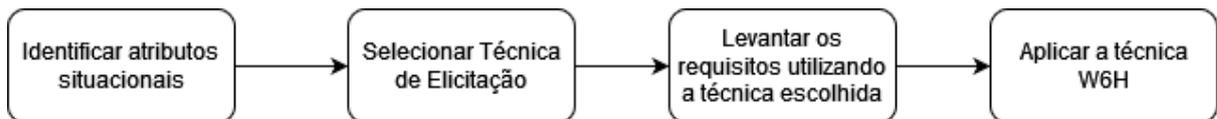
Esse documento  importante devido ao fato do mesmo ser utilizado durante todo o processo de desenvolvimento, j que  um documento-vivo que visa reduzir o retrabalho em etapas posteriores, evitando falhas no projeto de software, alm de ajudar a estimar os custos, perigos e cronogramas do produto.

2.4 SRAAF

Segundo Kumar e Deraman (2019), o framework SRAAF proposto tem como objetivo auxiliar os engenheiros de requisitos na elaborao dos requisitos claros e evidentes do sistema ajudando tambm na escolha da tcnica de elicitaco adequada levando em considerao alguns atributos e aplicando a tcnica W6H em cada requisito levantado. O framework prop abordar a questo da ambiguidade nos requisitos de uma maneira mais eficiente, ajudando assim na impreciso dos requisitos visando uma melhoria da qualidade do SRS, j que a ambiguidade  um dos principais problemas na etapa de especificao de requisitos. O framework consiste em 4 etapas: (i) Identificar os atributos situacionais necessrios para seleo da tcnica de elicitaco (ii) Selecionar a tcnica de elicitaco

adequada levando em consideração parâmetros situacionais associados ao projeto, domínio, partes interessadas e lista de técnicas de elicitação,(iii) Realizar levantamento dos requisitos (iv) Aplicação do W6H para remoção da ambiguidade nos requisitos que foram levantados (Figura 5).

Figura 5 - Etapas básicas do framework SRAAF



Fonte: Kumar (2019).

2.4.1 Identificação dos atributos situacionais

A primeira etapa consiste na identificação dos atributos situacionais. Esses atributos que serão levantados serão baseados no projeto em questão na qual será aplicado o framework. Esses atributos servirão posteriormente para selecionar a técnica de elicitação adequada para o projeto em questão.

2.4.2 Selecionar a técnica de elicitação adequada

A etapa para determinar os parâmetros significativos são selecionados 19 parâmetros, onde esses parâmetros são divididos em 3 categorias: Projeto, engenheiro de requisitos e partes interessadas. Podemos visualizar uma dessas categorias como mostrado na Figura 6.

Figura 6 - Parâmetros significativos do projeto

Tamanho	Complexidade	Requerimento Incerteza	Tempo Restrição	Atingível Em formação	Projeto Modelo	Processo Tempo	Tipo de em formação
Pequena	Baixo	Baixo	Baixo	Nada	Novo	Começo	Elementar
Médio	Médio	Médio	Médio	Menor	Existir	Intermediário	Importante
Grande	Alto	Alto	Alto	Mais alto		Estágio final	Crítico

Fonte: Kumar (2019).

O framework possui algumas técnicas de elicitação a serem utilizadas para a seleção, essas técnicas principais são o questionário, entrevista, casos de uso, histórias de usuário, prototipagem, JAD, RAD, etnografia.

Depois disso são identificados os parâmetros situacionais do projeto a ser utilizado,

são verificados os valores dos parâmetros significativos apresentados anteriormente para serem mapeados a fim de selecionar a técnica de elicitação mais apropriada. Logo em seguida é feito um mapeamento dos parâmetros situacionais juntamente com os parâmetros significativos, levando em consideração os valores dos atributos significativos com as diferentes técnicas de elicitação que é definida pelos analistas de requisitos. A figura 7 apresenta esses valores para a categoria de (engenheiro de requisitos), por exemplo. Portanto, após esse mapeamento é possível que o engenheiro de requisitos consiga selecionar a técnica mais apropriada para o projeto.

Figura 7 - Valores dos atributos significativos (engenheiro de requisitos)

Atributos	Experiência de Elicitação				
	Experiência com ET	Treinamento formal	do Engenheiro de Requisitos Novato	Experiência de domínio	Problema Compreensão
	Zero Baixo Alto	Baixo Médio	Alto	Experiente Proficiente	Iniciante experiente Proficiente
Questionários	Z, L, H	L, H	E, P	E, P	P, C
Entrevista aberta	L, H	H	E, P	N, E, P	P
Entrevista estruturada	L, H	L, H	E, P	E, P	C
Debate	L, H	L, H	P	N, E, P	P, C
Observação de Tarefa	Z, L, H	L, H	N, E, P	N	P, C
Análise de Documentos	H	H	E	E	C
Análise de protocolo	L, H	H	P	N, E	C
Observação do participante	L, H	L, H	N, E, P	N, E	C
Grade de repertório	L, H	H	E, P	E, P	C
Grupo nominal	Z, L, H	L, H	N, E, P	E, P	P, C
Técnica Delphi	L, H	H	N, E, P	E, P	P, C
Cenários/casos de uso	L, H	H	E, P	E	C
Prototipagem	L, H	H	E, P	P	P, C
Amostra	L, H	H	E, P	E, P	P, C
Workshop JAD/RAD	L, H	H	P	E, P	P

Fonte: Kumar, 2019

2.4.3 Aplicação do W6H

Por fim, após selecionada a técnica e feito todo o levantamento de requisitos utilizando-se da técnica de elicitação é então aplicada a técnica W6H para remover ambiguidades, além da possibilidade de enriquecer os requisitos elicitados com mais informações. Onde é feita a documentação e revisão dos requisitos levantados.

Por fins deste estudo, a utilização da técnica W6H se torna importante pelo fato da documentação de requisitos ser de extrema relevância nas próximas etapas do projeto, pois a aplicação do framework SRAAF gera um possível impacto nas próximas etapas do projeto.

2.5 W6H

A elicitação de requisitos é uma prática importante na ER, já que a ambiguidade está presente na descrição, entendimento e interpretação das necessidades humanas. O surgimento dessa ambiguidade se dá na comunicação entre as partes interessadas e o analista de requisitos, já que dependem drasticamente da linguagem e da linguística. Contudo, é importante que ambos utilizem uma linguística adequada e correta para que a comunicação seja clara, e os requisitos estejam de acordo com o que é necessário (MUJAHID, 2015).

A técnica W6H é uma técnica que se utiliza de perguntas interrogativas (which, where, who, why, when e how) para elicitação de requisitos. Essas perguntas interrogativas são divididas de acordo com o ponto de vista das partes interessadas (usuários, desenvolvedores, etc...) para facilitar a análise dos requisitos elicitados.

A divisão contém as preocupações do ponto de vista de um determinado grupo de partes interessadas que podem ser satisfeitas fazendo a pergunta interrogativa correspondente. Por exemplo: Que sistema será implantado e onde? Quais os dados que este sistema (iteração/lançamento) utilizará?. Essas perguntas são colocadas de forma ordenada fazendo com que a pergunta a ser feita pelo engenheiro de requisitos não gere qualquer dúvida, diminuindo assim a incerteza durante o processo.

Tanto a elicitação de requisitos quanto a ponte de lacunas são tarefas desafiadoras generalizadas. A técnica W6H oferece uma abordagem estruturada para enfrentar esses desafios. Essa técnica pode ser facilmente integrada aos processos e projetos de levantamento de requisitos reais, já que melhora a coleta de informações durante levantamento de requisitos e análise de requisitos, na qual será utilizada neste presente trabalho.

2.6 Desenvolvimento distribuído de software (DDS)

Nos últimos 10 anos devido ao efeito da globalização, as empresas de engenharia de software adquiriram cada vez mais o uso de formas distribuídas tanto nacionalmente como internacionalmente para realizar projetos de desenvolvimento de software, na qual se tornou cada vez mais popular que projetos de software sejam formados por equipes que trabalham de forma geograficamente distribuídas (PIRI, 2008). De acordo com Prikladnicki et. al (2011), o DDS é nada mais do que equipes de pessoas que se organizam de forma distribuída pelo mundo com o objetivo de se produzir um software, na qual as empresas cada vez mais estão utilizando o DDS.

Hoje existem vários motivos que fizeram com que as empresas de desenvolvimento de software começassem a adotar este modelo de trabalho. De acordo com Zapata, alguns desses motivos são a redução de custos de desenvolvimento, melhor aproveitamento global da escassa disponibilidade de mão de obra, maior número de horas de trabalho à disposição, incentivos ao investimento em mercados emergentes e crescimento considerável da demanda global (ZAPATA, 2012).

Devido à escassez de mão de obra e ao aumento das empresas estrangeiras em busca de profissionais altamente qualificados em outras regiões, muitas empresas têm enfrentado declaração de falência. Nesse cenário, o DDS (Desenvolvimento de Equipes a Distância) tem sido considerado uma solução alternativa para esse problema.

Sendo assim, este trabalho leva em consideração o ambiente do DDS, a fim de fornecer indícios e informações acerca do DDS para o uso do framework SRAAF, e se o mesmo gera algum impacto diante da aplicação e domínio do framework SRAAF.

3 TRABALHOS RELACIONADOS

Nesta etapa foram feitas as pesquisas e estudos a fim de levantar a importância deste trabalho. Para encontrar os trabalhos relacionados foram feitas pesquisas de assuntos específicos na busca, como por exemplo: engenharia de requisitos, desenvolvimento distribuído de software, técnicas de elicitação, e levantamento de Requisitos. Essas buscas foram utilizadas nas bases de dados do IEEE Explorer e ACM Library, onde a partir desses assuntos foram realizados estudos comparativos, na qual foi possível selecionar alguns artigos relevantes para o presente trabalho.

Segundo Zapata *et al.* (2012), o desenvolvimento distribuído de software que são conduzidas por equipes que trabalham de forma distribuída geograficamente tem crescido de forma desproporcional, devido a vários fatores que impulsionaram essa ideia (Zapata et al., 2012). O artigo de Zapata *et al.* (2012) traz uma proposta para o processo de desenvolvimento de software dentro da ER levando em consideração o ambiente *home-office*. Os autores apresentam um estudo comparativo da eficiência das técnicas de elicitação tradicionais (Entrevista, Questionário, Brainstorming) empregues a cenários distribuídos de desenvolvimento de software, tendo como um dos objetivos a melhoria da comunicação, sendo esse um dos principais pilares do presente trabalho.

O Zapata *et al.* (2012) verifica também qual técnica de elicitação tradicional é a mais eficiente, quando aplicada em desenvolvimento distribuído de software levando em consideração a comparação com desenvolvimento de software co-localizado, que são equipes que trabalham no mesmo local e horário. A pesquisa foi dividida em 2 execuções onde a primeira foi a execução aplicada em um contexto distribuído de software, e outra no contexto presencial.

Foram utilizados grupos de alunos que simularam equipes de elicitação, e os professores como papéis de clientes, onde para cada grupo foram atribuídas técnicas de elicitação a serem executadas nos dois contextos. O objetivo foi o de comparar os resultados obtidos, utilizando-se como base um documento de requisitos de software (DRS) produzido pelos alunos para avaliação da qualidade. No final do experimento foram avaliados e qualificados os documentos DRS produzidos a fim de se obter um indicador da eficácia da técnica de elicitação aplicada.

Por fim, através das comparações dos 2 contextos, os autores concluíram que as técnicas tradicionais utilizadas em ambientes distribuídos possuem uma ineficiência de 10% a menos se comparado a empresas que trabalham de forma presencial. Esse fato pode-se dar devido ao impacto que as ferramentas tecnológicas de comunicação possuem em um processo que envolve muita comunicação, neste caso o processo de elicitação de requisitos.

Este trabalho de conclusão tem como diferencial a aplicação de um framework SRAAF que existe no mercado de trabalho visando uma melhoria de qualidade para SRS. O trabalho de Zapata et al. (2012) apresenta um estudo sobre as técnicas de elicitação aplicadas no DDS, contudo não aplica nenhuma técnica visando melhoria do SRS, bem como o estudo realizado não contemplou a indústria.

Lloyd *et al.* (2002) afirma que as equipes de desenvolvimento de software são frequentemente distribuídas geograficamente a partir de seus clientes e usuários finais. Isso cria desafios significativos de comunicação e coordenação que afetam a eficácia da engenharia de requisitos (LLOYD; ROSSON; ARTHUR, 2002). O trabalho de Lloyd *et al.* (2002), tem como objetivo realizar um estudo que auxilia a ER distribuída para um projeto de desenvolvimento de software, além de buscar identificar quais técnicas de elicitação de requisitos funcionam melhor no contexto distribuído. Neste trabalho foram apresentadas algumas técnicas de elicitação mais usadas atualmente na engenharia de requisitos como a entrevista, casos de uso, brainstorming, prototipagem que serão consideradas no estudo de caso deste trabalho. No trabalho Lloyd *et al.* (2002) foram divididos 6 grupos de pessoas de forma distribuída ramificadas entre analista de requisitos e partes interessadas, onde durante o processo foi necessário produzir o documento SRS de forma distributiva que seria posteriormente avaliado, na qual seria aplicado um conjunto de métricas necessárias para avaliar a qualidade dos SRS produzido pelos 6 grupos. O trabalho foi dividido em 2 propósitos principais que são:

- Identificar motivos que levaram os grupos a escrever o documento SRS com qualidade;
- Avaliar a eficiência das técnicas de elicitação de requisitos usadas na forma distribuída;

Foram avaliados os fatores que poderiam ter afetado a qualidade do SRS elaborado, além da eficácia das técnicas de elicitação que foram utilizadas no trabalho para a coleta dos requisitos. Através de uma média de eficácia pode-se concluir que algumas técnicas de

elicitação são mais eficientes do que outras em uma ER distribuída, contudo houve uma relação negativa entre a eficácia da técnica de elicitação de requisitos e a qualidade geral do SRS. Porém somente a técnica de prototipagem e questionários foram avaliadas de forma negativa. As técnicas de brainstorming, gerenciamento de requisitos e casos de uso foram avaliadas como as mais eficazes. O resultado sugere que a participação ativa das partes interessadas envolvidas de forma síncrona se torna mais eficaz no processo de ER distribuída, do que de forma assíncrona.

O trabalho de Lloyd *et al.* (2002) traz um estudo sobre a eficácia de algumas técnicas de elicitação no formato remoto e causas que afetam a qualidade do DDS. Apesar do trabalho frisar a melhoria da qualidade do DDS, ele não traz essa aplicação no mercado de trabalho no contexto *home-office* e também não trata da aplicação do SRAAF ou qualquer outro framework.

No estudo de Hickey e Davis(2003) foi realizada uma pesquisa com os analistas que atuam no mercado de trabalho com o intuito de saber como os mesmos selecionam as técnicas de elicitação a serem usadas no projeto de desenvolvimento de software. Foram usados como base para a pesquisa 9 analistas de requisitos experientes, com a finalidade de se obter os resultados mais claros. Para tanto, foi adotada uma abordagem qualitativa através de 3 métodos (experiência do analista, análise dos documentos, entrevista com os analistas) para a coleta dessas informações visando definir características que afetam a seleção da técnica de elicitação.

A realização das entrevistas tiveram como objetivo entender como são selecionadas as técnicas por parte dos analistas, na qual essas informações foram coletadas e documentadas a fim de gerar uma lista de características situacionais para serem categorizadas e analisadas. Pode-se concluir que para selecionar a técnica existe um conjunto de características situacionais que levam na sua escolha, além de um conjunto de habilidades básicas e adicionais necessárias, não só para seleção da técnica, mas também para sua própria realização. O resultado da pesquisa não obteve conclusões muito ricas, mas ajudou a compreender as técnicas, por exemplo, a etnografia que foi destacada como a mais eficaz, e ajudou a levantar quais fatores situacionais os analistas consideram para escolher a técnica.

O trabalho de Hickey e Davis(2003) apesar de trazer um estudo de como os analistas selecionam a melhor técnica para cada projeto, o mesmo não aborda o contexto do DDS e nem uma aplicação de uma técnica ou framework. No caso deste trabalho aborda tanto a

escolha de uma técnica de elicitação de requisitos eficaz, como também a aplicação do W6H para melhoria da qualidade do SRS.

De acordo com Kausar *et al.* (2010), um dos principais motivos que contribuem para falhas em projetos de software são os requisitos pouco claros e imprecisos. Seu trabalho apresenta orientações para seleção de técnicas de elicitação de requisitos como entrevista, histórias de usuário, etnografia, cenários, protótipo levando em consideração diferentes aspectos de determinado projeto, além de procedimentos de seleção combinatória de técnicas de elicitação, que podem favorecer o sucesso do projeto, já que devido a falta de compreensão das técnicas por parte do engenheiro de software pode acarretar em uma escolha inapropriada da técnica de elicitação para o projeto. A pesquisa de Kausar *et. al* (2010) tem como objetivo proporcionar uma direção, para que a técnica a ser selecionada se adeque às necessidades dos engenheiros de requisitos.

No primeiro passo são descritas algumas orientações teóricas(parâmetros) que são utilizadas como base no projeto. No segundo passo foi proposto uma fórmula matemática em que são consideradas nove técnicas de elicitação mais comuns, na qual cada uma dessas técnicas são comparadas com os parâmetros levantados no primeiro passo, através de alguns pesos definidos para cada parâmetro que podem ser modificados de acordo com o projeto. O artigo apresentado não leva em consideração partes interessadas distribuídas geograficamente, o que já gera um ponto positivo se comparado com o objetivo deste trabalho. Como resultado foi obtido que o workshop de requisitos teve a maior pontuação juntamente com a técnica de entrevista e questionário logo em seguida.

O trabalho proposto por Kausar *et. al* (2010) apesar de trazer um critério que ajuda na seleção da técnica de elicitação, a mesma não leva em consideração a aplicação do critério no DDS, e nem envolve a aplicação de nenhuma abordagem de melhoria do documento SRS.

Segundo Kumar e Deraman (2019), a qualidade dos requisitos está diretamente relacionada à técnica escolhida pelo engenheiro de requisitos. No trabalho de Kumar (2019) é apresentado uma sugestão de framework conhecida como SRAAF que foi apresentado na seção 2.4. Além de ajudar na escolha da técnica através de alguns parâmetros situacionais, a mesma sugere um procedimento que auxilia a evitar ambiguidades nos requisitos levantados utilizando-se da técnica W6H descrito na seção 2.5 , evitando assim que a mesma chegue até o documento SRS. O SRAAF é dividido em 2 fases: Seleção da técnica de elicitação com base em atributos situacionais, a segunda é a aplicação da técnica W6H para evitar a ambiguidade nos requisitos. O trabalho conclui que a seleção da técnica de elicitação correta é

muito importante, pois um levantamento inadequado dos requisitos pode levar a resultados indesejados, com isso o SRAAF auxilia os Engenheiros de Requisitos na escolha da técnica aumentando assim o sucesso de projetos futuros, já que na sua maioria não sabem quando usar uma determinada técnica de elicitação.

Por último o trabalho de Kumar e Deraman (2019) apesar de abordar todos os pontos apresentados em outros trabalhos relacionados acaba se diferenciando deste trabalho pelo fato do mesmo não aplicar o SRAAF no DDS. Na qual o framework SRAAF será aplicado no presente trabalho utilizando o DDS.

Na Tabela 1 é possível visualizar a relação entre os trabalhos relacionados juntamente com o presente trabalho. Os trabalhos apresentados focam principalmente nas diferentes técnicas de elicitação de requisitos existentes. Os trabalhos de Zapata *et al.* (2012), Lloyd *et al.* (2002) enfatizam o uso da eficácia de algumas técnicas de elicitação aplicadas no desenvolvimento de software distribuído, na qual é um dos objetivos deste presente trabalho. Já no trabalho de Hickey e Davis (2013), apesar de citar sobre as técnicas de elicitação e como os engenheiros de requisitos escolhem, não é apresentado um modelo que ajude na escolha da técnica. Já o trabalho de Kausar *et al.* (2010) apesar de apresentar um método que auxilie na escolha da técnica, a mesma não é apresentada dentro do contexto do DDS, já que os parâmetros apresentados referentes ao projeto não levam em consideração equipes que trabalham de forma remota, fazendo com que esses valores possam vir a sofrer mudanças se aplicadas nesse contexto.

O trabalho de Kumar (2019), ainda que apresente a aplicação de uma metodologia focada na seleção da técnica de requisitos e para lidar com a ambiguidade no SRS englobando todos os pontos relevantes, o mesmo não é aplicado no contexto do DDS. Podemos levar em consideração que as pesquisas aqui apresentadas se diferenciam do presente trabalho, pelo fato de não focarem em um estudo de caso levando em consideração uma abordagem que ajude na escolha da técnica de elicitação no contexto distribuído.

Tabela 1 – Trabalhos Relacionados

Assuntos Abordados	Zapata et al. (2012)	Lloyd et al. (2002)	Hickey; Davis (2003)	Kausar et al. (2010)	Kumar; Deraman (2019)	Este Trabalho
Consideram Técnicas de Elicitação	x	x	x	x	x	x
Realiza pesquisa sobre Engenharia de Requisitos	x	x			x	x
Aplica abordagem/técnica para ajudar na escolha da técnica de elicitação				x	x	x
Considera o DDS	x	x				x
Ajudar a melhorar o documento SRS		x			x	x
Realiza pesquisa sobre Elicitação de Requisitos	x	x	x	x	x	x

Fonte: O autor, 2022

4 PROCEDIMENTOS METODOLÓGICOS

Este capítulo apresenta as atividades realizadas para alcançar os objetivos propostos (ver Figura 8).

Figura 8 - Procedimentos metodológicos



Fonte: O autor (2022).

4.1 Preparação do ambiente para aplicação do framework

Nesta etapa foi realizada toda a preparação do ambiente na qual o framework foi aplicado e conduzido pelo autor deste trabalho, uma vez que este atualmente exerce o cargo de PO(Product Owner) na empresa onde o estudo de caso foi executado. Foi realizada uma reunião via google meet juntamente com a equipe participante para selecionar o projeto em si para o estudo de caso. Através de uma análise de viabilidade se seria possível aplicar o framework, na qual foi escolhido um projeto que tem como princípio a venda de produtos mercadológicos por Whatsapp. Esse projeto foi escolhido pelo fato do mesmo ser um projeto em fase inicial de descoberta, o que viabiliza o uso do framework.

A empresa na qual foi aplicada o framework é uma empresa do ramo de e-commerce no setor de varejo/alimentos. A mesma adota atualmente o formato *home-office* desde o início da pandemia da Covid-19 em 2020. A empresa trabalha com o desenvolvimento de interfaces tanto mobile como Web para maior flexibilidade para seus clientes, onde a empresa adota como metodologia ágil o Scrum. O projeto a ser utilizado tem como objetivo a aplicação do SRAAF.

A empresa onde foi realizado o estudo de caso possui o contexto necessário no desenvolvimento distribuído de software. Apesar de sua sede se localizar na cidade de Belo Horizonte-MG, a mesma possui ao todo 97 colaboradores espalhados entre os diferentes

estados do Brasil, na qual em sua maioria se concentram nos estados de Minas Gerais, São Paulo, Goiás e Rio Grande do Sul.

Em conversa com alguns dos colaboradores visando o levantamento de informações, foi relatado a questão de funcionalidades que acabam tendo um duplo entendimento por parte dos analistas e desenvolvedores da empresa, na qual os requisitos acabam às vezes sendo incompletos dependendo da sua complexidade. O que culmina em uma nova especificação ocasionando um atraso no prazo de entrega do projeto, já que de acordo com Hickey e Davis(2003) esse problema se dá pelo fato da dificuldade de se escolher uma melhor técnica de requisitos. Os colaboradores que se propuseram a participar do estudo se mostraram entusiasmados a oferecer qualquer tipo de apoio que pudesse agregar alguma relevância para o processo, expondo as suas opiniões e sugestões ao framework que foi empregado neste estudo de caso.

4.2 Estudo de caso

Esta fase teve como propósito realizar um estudo de caso exploratório com objetivo de se descobrir acontecimentos que ocorrem durante a aplicação do framework SRAAF, a fim de se gerar possíveis melhorias, além de constituir possíveis conceitos para o framework em um ambiente real. O estudo de caso foi realizado com a aplicação do SRAAF no grupo de desenvolvimento da “EmpresaV” em um projeto no contexto home-office.

4.3 Análise dos resultados com aplicação do framework

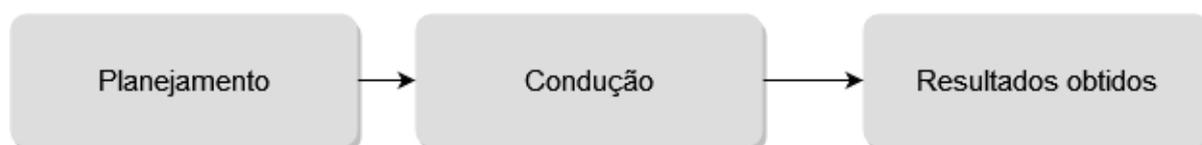
Os resultados que foram obtidos no estudo de caso foram avaliados de forma quantitativa, onde as respostas foram gravadas e apresentadas na forma de tabela e gráficos, além do estudo dos artefatos que foram gerados através do framework, daquela forma como também qualitativa por meio de método Grounded Theory apresentado por Barbosa(2017).

Na qual possibilitou realizar uma análise comparativa desses resultados, possibilitando uma análise mais precisa da eficácia do framework no contexto home-office. Por fim, com a abordagem já validada, buscou-se atingir os objetivos que foram propostos na pesquisa, para decidir como o framework proposto abrandou de uma forma mais completa o seu uso na ER, levando em consideração os objetivos específicos expressos neste trabalho.

5 ESTUDO DE CASO

Neste capítulo são apresentadas todas as etapas usadas para o estudo de caso utilizadas para entendimento da pesquisa no contexto apresentado anteriormente. De acordo com Viana (2021), um estudo de caso é nada mais do que uma estratégia que visa o aprofundamento em uma pesquisa com o intuito de se investigar benefícios e possíveis falhas na pesquisa, onde através desse estudo é possível coletar e analisar os dados de forma mais extensiva.

Figura 9 - Etapas do estudo de caso



Fonte: O autor (2022).

5.1 Planejamento da aplicação do processo no ambiente remoto

Para o planejamento da aplicação do processo em um ambiente remoto para o estudo de caso foi escolhida uma equipe que trabalha de forma contratual na “empresaV”. O time é composto por 5 pessoas que foram divididas basicamente por um PO(Product Owner), três analistas de requisitos e dois QAs(*Quality Assurance*) que constituem parte da empresa. Este time está distribuído entre os estados do Ceará, Minas Gerais e São Paulo. O time foi basicamente as partes interessadas responsáveis pela aplicação do framework SRAAF.

5.2 Condução do estudo de caso

Nesta etapa será descrita a aplicação do framework na “empresaV” pela equipe, com o propósito de se obter dados necessários para análise dos resultados obtidos.

O autor deste estudo de caso desenvolveu um questionário(Ver Apêndice A) que foi utilizado para coletar os dados para a análise do framework, o trabalho de (FIORI;VITÓRIA, 2022) serviu de fundamento da questão e principal fonte de referência para elaboração do questionário. Seu objetivo foi avaliar o uso do framework SRAAF pelo time por meio de uma análise qualitativa e quantitativa.

A execução do SRAAF foi dividida em 3 etapas(ver Tabela 2), na qual o processo

utilizado foi seguido de acordo com que foi apresentado por Kumar(2019).

Tabela 2 - Descrição das Etapas para execução do SRAAF

	Descrição
1º etapa	Esclarecimento do framework;
2º etapa	Aplicação do framework SRAAF;
3º etapa	Aplicação do questionário para equipe.

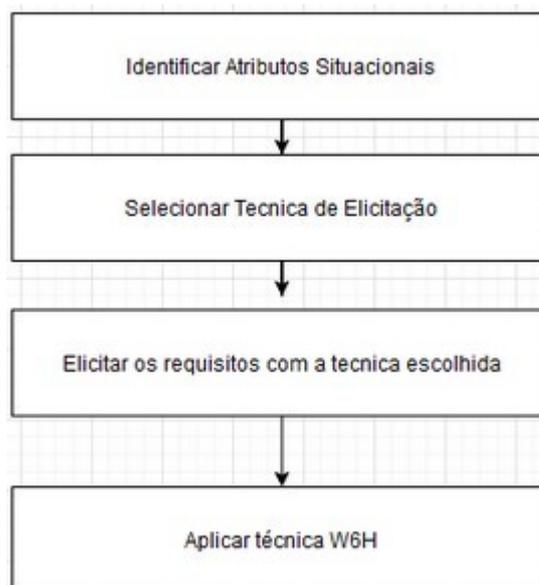
Fonte: O autor (2022).

Na 1º etapa ocorreu a explicação para a execução do framework, como seus objetivos, técnicas que serão utilizadas, e os resultados que se busca esperar com sua aplicação, além de apresentar o passo a passo de como será feito o andamento do framework.

Na 2º etapa foi realizada a aplicação do framework SRAAF (ver Figura 10). Essa aplicação seguiu todos os passos apresentados por Kumar (2019).

Já a 3º etapa consiste da aplicação do questionário para a equipe. O questionário será aplicado remotamente após a execução do framework a fim de extrair informações sobre o uso do framework SRAFF pela equipe, como o feedback e coleta de dados acerca da experiência, além de dados relevantes que serão utilizados para o estudo, como problemas e dificuldades encontradas na aplicação.

Figura 10 - Processo do SRAAF



Fonte: Kumar (2019).

5.3 Análise dos resultados obtidos

Nesta fase são apresentados os resultados obtidos com a utilização do framework SRAAF. Essa análise foi realizada por meio de diretrizes de análise qualitativa e quantitativa, com o objetivo de examinar a eficácia do SRAAF por meio dos participantes do estudo e levando em consideração suas experiências com a aplicação.

5.3.1 Análise quantitativa do uso do framework

Um questionário foi usado para coletar esses dados com a intenção de avaliar a experiência dos participantes do projeto. A pesquisa foi dividida em duas seções, a primeira focada nas experiências dos entrevistados usando o framework SRAAF (Software Requirement Ambiguity Avoidance Framework) e a segunda discutindo os desafios associados ao uso do framework SRAAF. Os dados foram descritos e analisados quantitativamente. Este questionário encontra-se no Apêndice A.

Na figura 11 abaixo é possível visualizar o gráfico com os resultados relacionados a etapa que consiste no uso do framework SRAAF na etapa da engenharia de requisitos. De acordo com a equipe que participou do processo, os participantes consideraram que a utilização do framework SRAAF melhoraria a etapa da ER. Já na questão seguinte apresentada no gráfico, a maioria dos participantes afirmaram que o framework melhora a qualidade dos requisitos que foram levantados durante o processo. No entanto, um participante discorda parcialmente que possa haver uma melhoria na qualidade desses requisitos. Esse fator pode ser decorrente da falta de domínio do projeto.

Em resposta à terceira pergunta, os participantes concordaram que o SRAAF ajudaria a melhorar a qualidade do documento SRS. Na quarta questão para a maioria dos participantes concordaram que o framework diminuiu a ambiguidade nos requisitos que foram levantados, onde apenas um participante discordou de forma parcial que houve uma redução na ambiguidade dos requisitos. Na quinta afirmação apenas dois participantes concordaram que o framework possibilita escolher a melhor técnica de elicitação para o projeto em questão, enquanto a maioria dos participantes discordou um pouco. Essa discordância pode ter sido causada pela dificuldade de compreensão do framework, apesar de a técnica escolhida já ser bastante conhecida e utilizada pelos analistas da "empresaV".

Por último, mas não menos importante, a maioria dos participantes acredita que a estrutura é útil durante a fase de engenharia de requisitos, embora um participante discorde de qualquer uma das opções se mantendo de forma neutra. O fato de haver participantes que

não atuavam diretamente na área de requisitos, no caso os QAs, pode ter sido a motivação.

Assim sendo, podemos concluir que o framework SRAAF é útil na etapa da engenharia de requisitos, porém o framework ajudou na escolha efetiva da melhor técnica de elicitação de requisitos a ser utilizada para o projeto e a redução da ambiguidade dos requisitos, como também se apresentou eficaz para construção do documento SRS.

Figura 11 - Experiência com o framework SRAAF



Fonte: O autor, 2022

5.3.2 Análise qualitativa do uso do framework

Através desta seção serão apresentados os resultados da análise qualitativa através das respostas obtidas por meio do questionário, na qual foi analisada a aplicação do processo do framework. As perguntas que foram feitas aos colaboradores se encontram no Apêndice B.

Foi necessário o uso da grounded theory para realizar a análise qualitativa (Grounded theory) proposto por Barbosa. Segundo Barbosa (2017), o objetivo da GT é gerar uma teoria como uma coleção de hipóteses conectadas que são geradas por meio da comparação contínua de dados em níveis crescentes de abstração.

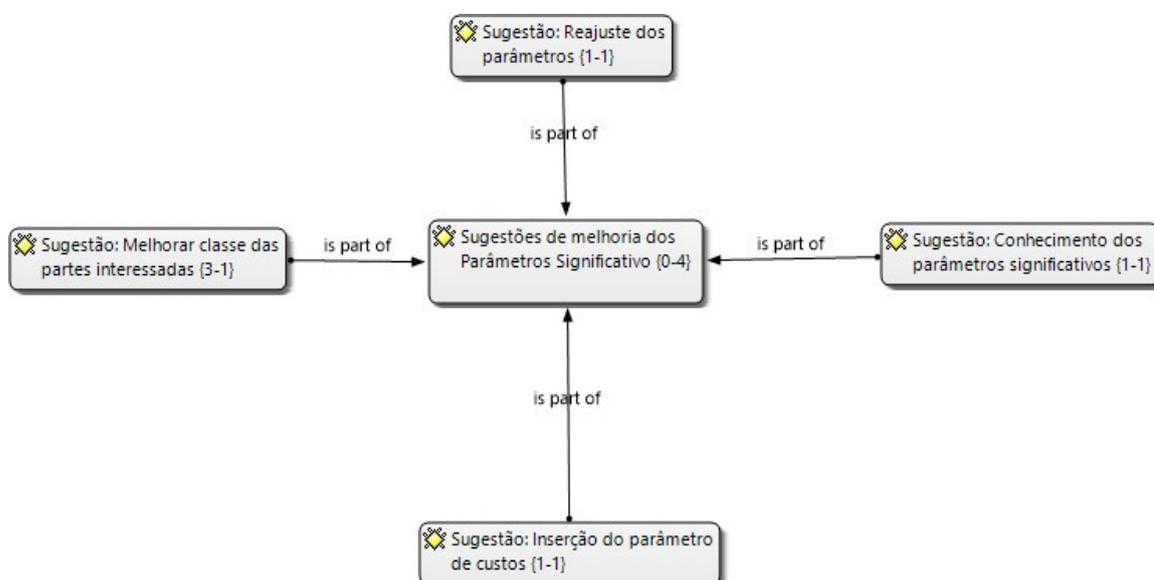
Barbosa apresenta três tipos diferentes de codificação: axial, aberta e seletiva, mas para este estudo de caso foi utilizada apenas a codificação aberta, que é a mais adequada para este estudo de caso. Após análise e comparação dos dados, os conceitos foram agrupados em

categorias nomeadas pelo autor do trabalho. Nessas categorias, os códigos foram gerados com base nas perguntas feitas na pesquisa para serem relacionadas às categorias. Além disso, o software atlas.ti foi utilizado para analisar e gerenciar os dados coletados. Abaixo estão as categorias que foram geradas:

- Sugestões de melhoria dos Parâmetros Significativos;
- Técnicas de Elicitação levantadas no framework;
- Dificuldades na aplicação da técnica de elicitação;
- Características do framework SRAAF que auxilia na etapa da engenharia de requisitos;
- Desafios/dificuldades da utilização do framework SRAAF.

Após identificar as categorias com seus respectivos códigos foram geradas redes que fazem para compor o network dos códigos de cada categoria. Essa primeira rede pode ser vista na Figura 12.

Figura 12 - Rede de sugestões de melhoria dos parâmetros significativos



Fonte: O autor, 2022

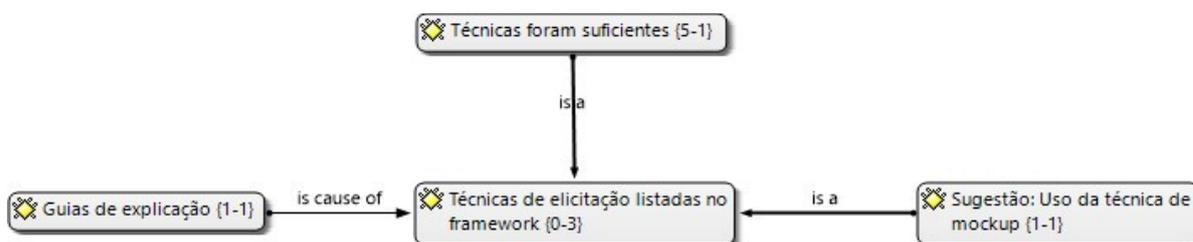
Na imagem é mostrada a rede que foi criada usando os códigos gerados pelas respostas dos participantes a uma pergunta enviada por meio do questionário. A indagação feita aos participantes teve como objetivo determinar se os parâmetros significativos

apresentados pelo framework são suficientes para o processo. Embora todos afirmaram que os parâmetros eram suficientes, algumas observações adicionais foram feitas, incluindo sugestões de melhoria. Podemos destacar que os códigos que tiveram uma maior relevância foram os códigos de melhorar a classe das partes interessadas e inserção do parâmetro de custos. Mais detalhes podem ser vistos na Tabela 3.

Tabela 3 - Códigos: Sugestões de melhoria dos Parâmetros significativos

Categoria	Códigos	Colaborador	Resposta
Dificuldades para seleção da técnica de elicitação de requisitos.	Sugestão: Melhorar classe das partes interessadas	C1,C2,C5	C1- <i>Sim, não senti falta de nenhum parâmetro. Porém precisaria definir melhor a classe de partes interessadas de quem seriam essas partes.</i> C2- <i>[...] não ficou claro quem seria as partes interessadas dos parâmetros.</i> C5- <i>Sim, só não ficou claro sobre a classe partes interessadas, pois não entendi muito bem qual seria essas partes. Além disso, algum dos valores dos parâmetros não deixou claro o que significava.</i>
	Sugestão: Reajuste dos parâmetros	C2	C2- <i>Sim, mas creio que os parâmetros podem ser reajustados de acordo com tipos específicos de projetos[...].</i>
	Sugestão: Inserção do parâmetro de custos	C4	C4- <i>Foram suficientes sim, porém valeria a pena inserir um parâmetro situacional voltado ao projeto, no caso um parâmetro sobre custos(baixo, médio, alto).</i>
	Sugestão: Conhecimento dos parâmetros significativos	C3	C3- <i>Creio que foram suficientes. Mas o interessante seria que eu como engenheiro de requisitos tenha um conhecimento dos parâmetros situacionais antes de utilizar a técnica, pois tinha parâmetros que eu não sabia o que significava.</i>

Figura 13 - Rede de técnicas de elicitação listadas no framework



Fonte: O autor, 2022

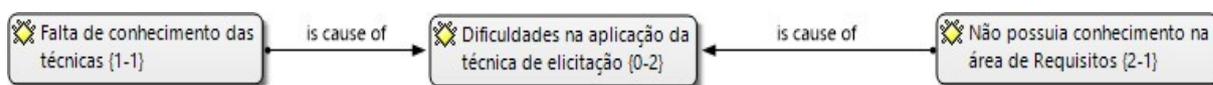
A Figura 13 apresenta a rede de network na qual teve como objetivo saber se as técnicas de elicitação apresentadas no framework foram suficientes e quais poderiam ser incluídas. Podemos destacar que os códigos que tiveram uma maior relevância foram os códigos: técnicas foram suficientes e guias de explicação. Mais detalhes podem ser vistos na Tabela 4.

Tabela 4 - Códigos: Técnicas de elicitação listadas no framework

Categoria	Códigos	Colaborador	Resposta
Técnicas de elicitação listadas no framework	Técnicas foram suficientes	C1,C2,C3,C4,C5	C2- <i>Sim foram suficientes.</i> C4- <i>Sim, apesar de não conhecer todas elas.</i>
	Guias de explicação	C5	C5- [...] <i>creio que poderia conter um guia de explicação das técnicas menos utilizadas que foram apresentadas no framework.</i>
	Sugestão: Uso da técnica de mockup	C2	C2- <i>Porém como sou analista de requisitos eu acabo utilizando uma técnica que seria o mockup que dependendo do projeto eu necessito desenvolver telas, o que me ajuda a levantar requisitos que não consigo ver se eu aplicar outra técnica. Talvez valeria a pena ser inserido.</i>

Fonte: O autor, 2022

Figura 14 - Rede de dificuldades na aplicação da técnica de elicitação



Fonte: O autor, 2022

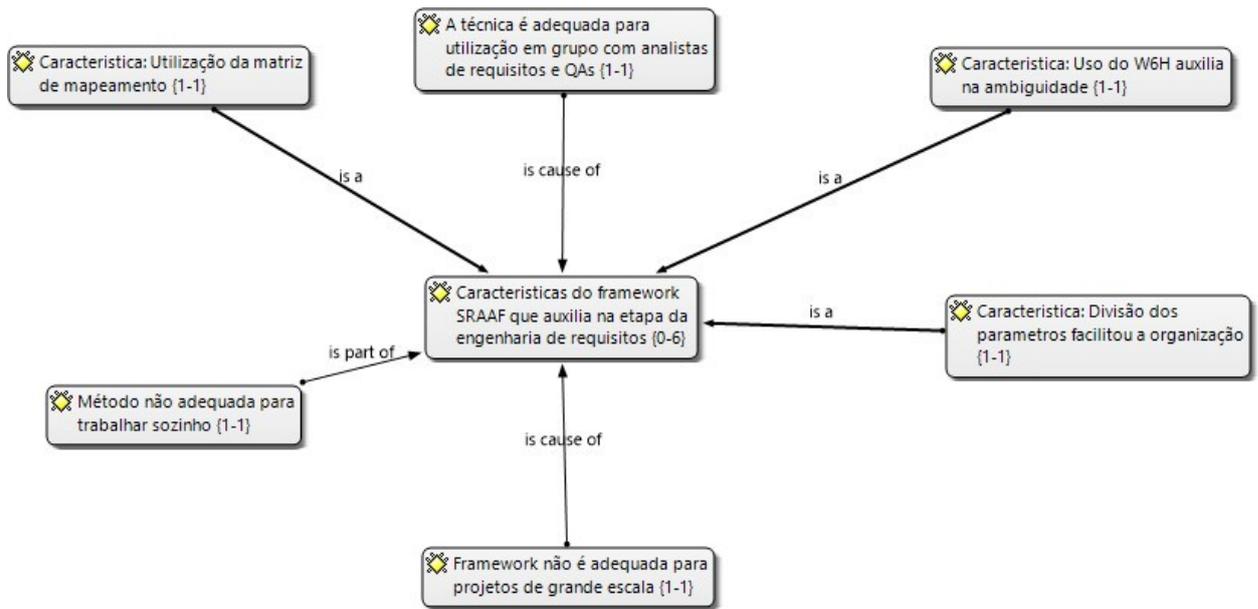
O objetivo da rede mostrada na figura 14 foi determinar se a equipe teve alguma dificuldade ao aplicar a técnica de elicitação. Devemos observar que os códigos com maior relevância foram aqueles que indicam falta de conhecimento técnico e falta de conhecimento na área de requisitos. Mais informações podem ser encontradas na Tabela 5.

Tabela 5 – Códigos - Dificuldades na aplicação da técnica de elicitação

Categoria	Códigos	Colaborador	Resposta
Dificuldades na aplicação da técnica de elicitação	Falta de conhecimento das técnicas	C4	<i>C4- Apesar de não conhecer todas as técnicas apresentadas, a que foi selecionado consegui aplicar sem problemas.</i>
	Não possuía conhecimento na área de requisitos	C2,C3	<i>C2- Senti dificuldade , pois não sou da área de requisitos. C3- Como não participo diretamente da área de requisitos acabei sentindo dificuldades.</i>

Fonte: O autor, 2022

Figura 15 - Rede de características do framework que auxilia na etapa da ER



Fonte: O autor, 2022

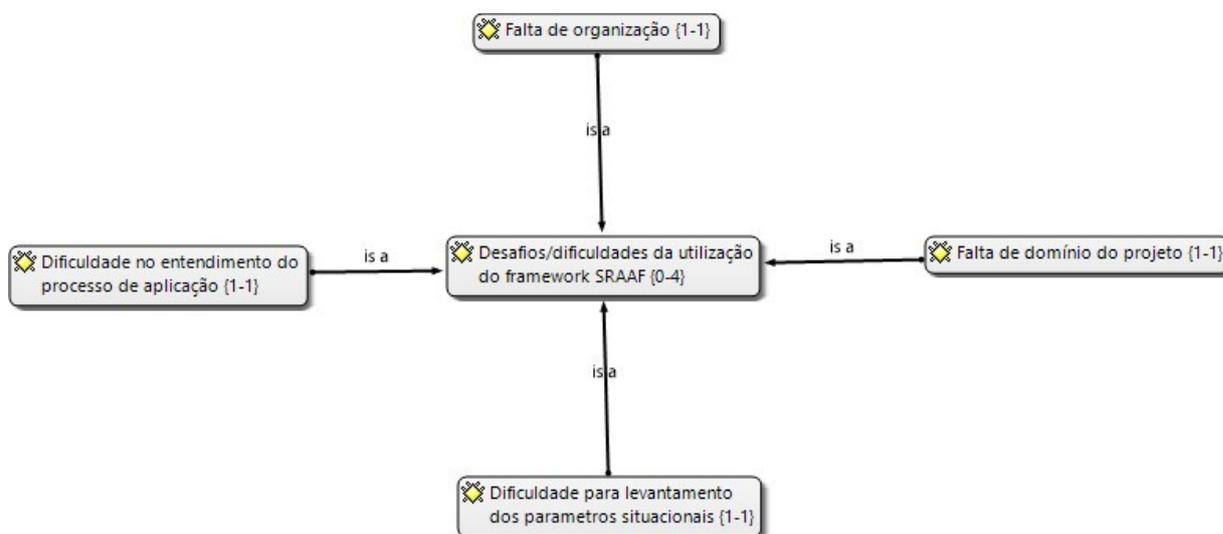
O objetivo da rede mostrada na Figura 15 foi avaliar quais características do framework auxiliavam na etapa da ER. Devemos observar que os códigos com maior relevância foram: Característica: Utilização da matriz de mapeamento, Característica: Uso do W6H auxilia na ambiguidade e a técnica é adequada para utilização em grupo com analistas de requisitos e QAs. A Tabela 6 pode ser consultada para maiores informações.

Tabela 6 – Códigos: Características do SRAAF auxilia na etapa da ER

Categoria	Códigos	Colaborador	Resposta
Características do SRAAF auxilia na etapa da ER	Característica: Utilização da matriz de mapeamento.	C1	<i>C1- A tentativa de resolver a ambiguidade é de extrema importância para os requisitos, pois evita que isso seja passado para os desenvolvedores de forma ambígua.</i>
	A técnica é adequada para utilização em grupo com analistas de requisitos e QAs.	C5	<i>C5- A utilização dele em grupo com outros analistas de requisitos e QAs ajuda a ter um mesmo direcionamento.</i>
	Característica: Uso do W6H auxilia na ambiguidade.	C3	<i>C3- A utilização do W6H para resolver ambiguidade auxiliou a deixar os requisitos mais completos.</i>
	Característica: Divisão dos parâmetros facilitou a organização.	C2	<i>C2- A divisão em 3 partes dos parâmetros significativos ajudou bastante na organização.</i>
	Framework não é adequado para projetos de grande escala.	C1	<i>C1- [...]. O que não ajudou é que ele não parece ser muito eficaz para projetos de grande escala, talvez para médios e pequenos projetos seja mais eficiente.</i>
	Método não adequado para trabalhar sozinho.	C5	<i>C5- [...]. Talvez para trabalhar sozinho, acho que não é um método adequado.</i>

Fonte: O autor, 2022

Figura 16 - Rede de Desafios/dificuldades da utilização do framework SRAAF.



Fonte: O autor, 2022

O objetivo da rede mostrada na Figura 16 foi apresentar os Desafios/dificuldades de elicitação da utilização do framework SRAAF.. Os códigos de maior significância foram: Característica: Falta de domínio do projeto, falta de organização e dificuldade no entendimento do processo de aplicação. A Tabela 7 pode ser consultada para obter informações mais detalhadas.

Tabela 7 - Códigos: Desafios/dificuldades da utilização do framework

Categoria	Códigos	Colaborador	Resposta
Desafios/dificuldades da utilização do framework SRAAF	Falta de organização	C4	<i>C4- Falta de organização para aplicar o framework.</i>
	Dificuldade para levantamento dos parâmetros situacionais	C2	<i>C2- Levantar os parâmetros situacionais foi uma etapa difícil no processo.</i>
	Falta de domínio do projeto	C3	<i>C3- Não conhecia praticamente nada do projeto em questão, o que me dificultou na hora de elicitar os requisitos de forma mais adequada.</i>

	Dificuldade no entendimento do processo de aplicação	C5	<i>C5- Consegui utilizar, porém senti dificuldades em entender o processo.</i>
--	--	----	--

Fonte: O autor, 2022

Levando em consideração a finalização da análise através do estudo e uso da GT, podemos concluir que o framework se mostrou adequado para os participantes do estudo, apesar de alguns pontos negativos terem surgido, como a falta de organização que acabou resultando em uma desafio na compreensão do processo. No entanto, também foram apontados alguns pontos positivos, como a utilização do W6H e a organização dos parâmetros significativos, que surgiram como um ponto crucial no processo. Observou-se também que foram sugeridos alguns pontos de melhoria que poderiam ajudar na assertividade do framework, o que poderia melhorar a experiência com o processo.

6 MELHORIA NO PROCESSO DO SRAAF

Durante o estudo de caso, foram descobertos alguns pontos de melhoria apontados pelos envolvidos no desenvolvimento do projeto. Uma das sugestões iniciais da equipe foi que a equipe precisava entender melhor a perspectiva do domínio sobre o uso do projeto para vendas via WhatsApp. Poderia ser mais fácil para os projetos que já se tenha um estudo sobre seu domínio por parte dos engenheiros de requisitos na qual possa facilitar a utilização do framework. Outro fator levantado foi que a equipe sentiu a necessidade de ter uma parâmetro significativo inserida para a área de custos. Isso é interessante porque os custos dos projetos são muito importantes, principalmente quando se trata de técnicas de elicitação que exigem a utilização de algumas ferramentas pagas, o que pode potencialmente aumentar a assertividade da seleção de técnicas eficazes.

Outro ponto observado foi que algumas pessoas tiveram dificuldade em aplicar a técnica de elicitação. Isso pode ser devido ao fato de que, embora todos os cinco participantes estivessem envolvidos no processo de descoberta dentro da "empresaV", nem todos ocupavam o cargo de engenheiro de requisitos. Um último ponto a se considerar pela equipe seria a falta de compreensão da classe das partes interessadas que compõem o framework, impossibilitando ou dificultando a fluidez do procedimento do SRAAF e resultando em um cronograma maior.

7 CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS

Este trabalho teve como objetivo investigar a utilização do framework SRAAF em equipes que trabalham de forma remota em um estudo de caso, onde teve como início um estudo de trabalhos correlacionados e posteriormente seu planejamento e aplicação do processo e avaliação dos resultados gerados visando trazer resultados obtidos com a aplicação do framework SRAAF na engenharia de requisitos.

A sua aplicação foi realizada entre cinco colaboradores que foram divididos em três analistas de requisitos e 2 QAs em uma empresa que atua no setor de E-commerce. Onde nenhum deles tem conhecimento de qualquer ferramenta que envolva a escolha da técnica de elicitação e ambiguidades nos requisitos. Na qual após a aplicação foi possível realizar uma pesquisa para coleta de informações e levantamento de feedbacks para entender a percepção por parte dos colaboradores sobre a experiência com o framework e se a mesma se mostrou útil para a ER. Os resultados que foram gerados trouxeram alguns pontos positivos como a redução da ambiguidade nos requisitos elicitados utilizando o W6H, o que ocasiona em uma melhoria no documento SRS, além de mostrar que o framework é útil e eficaz para os analistas de requisitos na etapa da ER.aperfeiçoamento dos parâmetros significativos que foi amplamente discutida pelos colaboradores durante a aplicação do framework.

Além disso também foram identificados alguns pontos negativos relacionados a melhoria framework que podem ser empregues no processo, como o sentido a falta de um guia de explicação, além de que a falta de organização acabou dificultando o processo da aplicação.

Trabalhos futuros avaliarão a viabilidade das melhorias sugeridas, principalmente quanto aos parâmetros para maior assertividade, e redesenho do processo do framework com o objetivo de aplicá-lo ao novo processo em questão. Este trabalho também examinará a eficácia da estrutura quando aplicada a projetos de grande escala, bem como o uso de várias técnicas em vez de apenas uma, conforme sugerido pela estrutura hoje.

REFERÊNCIAS

ABES. **Mercado Brasileiro de Software: Panorama e Tendências 2021**. 1ª ed. Associação Brasileira das Empresas de Software. São Paulo, 2021.

A.D. LIMA. **Formato de Especificação de Requisitos de Software (SRS)**. Disponível em: <https://comoprogramarjava.com.br/frameworks/java-base/formato-de-especificacao-de-requisitos-de-software-srs/>.

A.M. Hickey, A.M. Davis. **Elicitation technique selection: how do experts do it?** International Requirements Engineering Conference, 2003.

A. PIRI. **Challenges of Globally Distributed Software Development – Analysis of Problems Related to Social Processes and Group Relations**, 2008 IEEE International Conference on Global Software Engineering, 2008, pp. 264-268, doi: 10.1109/ICGSE.2008.33.

BARBOSA, M. W. 2017. Uma análise do uso de grounded theory em engenharia de software. **Revista Produção Online**. Disponível em: <https://www.producaoonline.org.br/rpo/article/view/2326>.

BRUM, Bruno Conde Perez; PENA, Leonardo. **Principais técnicas de levantamento de requisitos de sistemas. Engenharia de requisitos – Técnicas**. Disponível em: <http://brunobrum.wordpress.com/2011/04/27/principais-tecnicas-de-levantamento-de-requisitos-de-sistemas/>.

CHUA, D. V. Bernardo and J. Verner, "Understanding the Use of Elicitation Approaches for Effective Requirements Gathering," 2010 Fifth International Conference on Software Engineering Advances, 2010, pp. 325-330, doi: 10.1109/ICSEA.2010.89.

CARRIZO, Dante. **Comparison of Research and Practice Regarding What We Mean by "The Right Software Requirements Elicitation Technique"**. In: 2016 10th International Conference on the Quality of Information and Communications Technology (QUATIC). IEEE, 2016. p. 79-82.

CARVALHO, Pedro F. **Técnicas de Levantamento de Requisitos**. São José do Rio Preto, 2009. DAVID, Luiz. **Técnicas para reuniões em JAD (Joint Application Design)**. Disponível em: <http://engenhariadesoftware.info/downloads/JAD.ppt>.

CARMEL, Prentice-Hall, **Global software teams : collaborating across borders and time zones**, Nova Jersey, 1999.

FALBO, Ricardo de Almeida. **Análise de Sistemas: Notas de Aula**. Disponível em: <http://www.ceunes.ufes.br/downloads/2/mariateixeira-Notas%20de%20Aula.An%C3%A1lise%20de%20Sistemas.Prof%20Falbo.UFES.pdf>.

FABIANO, 2012. **Engenharia de Requisitos: Uma análise das técnicas de levantamento de requisitos**. Disponível em: http://professores.dcc.ufla.br/~terra/publications_files/students/2012_fumec_silva.pdf.

GARG, Nikita; AGARWAL, Pankaj; KHAN, Shadab. **Recent advancements in requirement elicitation and prioritization techniques**. In: 2015 International Conference on Advances in Computer Engineering and Applications. IEEE, 2015. p. 237-240.

GUPTA, Ashok & Deraman, Aziz. (2019). **A framework for software requirement ambiguity avoidance**. International Journal of Electrical and Computer Engineering (IJECE).

JOBS, Marc Fast. **JAD (Joint Application Design) – Técnica de levantamento interativo**. Disponível em: <http://vivenciandoti.blogspot.com.br/2010/03/jad-joint-application-design-tecnica-de.html>.

JAD Guidelines. **IT Project Management Certificate Program (v.0)**. Knowledge Structures, Inc. Disponível em: <http://www.ksinc.com/itpmcptools/JADGuidelines.pdf>.

J. VIANA. **Estudo de caso: o que é, exemplos e como desenvolver**. Disponível em: <https://keeps.com.br/estudo-de-caso-o-que-e-exemplos-e-como-desenvolver/>. Acesso em: 09 de out. de 2022.

KAUSAR; Saima Tariq; Saba Riaz; Aasia Khanum, (2010). **Guidelines for the selection of elicitation techniques**. In 6th International Conference on Emerging Technologies (ICET).

LLOYD, M. B. Rosson and J. D. Arthur, "Effectiveness of elicitation techniques in distributed requirements engineering," Proceedings IEEE Joint International Conference on Requirements Engineering, 2002, pp. 311-318, doi: 10.1109/ICRE.2002.1048544.

MISHRA, Deepti; MISHRA, Alok; YAZICI, Ali. **Successful requirement elicitation by combining requirement engineering techniques**. In: 2008 First International Conference on the Applications of Digital Information and Web Technologies (ICADIWT). IEEE, 2008. p. 258- 263.

MONITORA TEC. **Serviço de engenharia de requisitos: entenda como funciona** <https://www.monitoratec.com.br/blog/servico-de-engenharia-de-requisitos/>. Acesso em: 15 de jan de 2022.

M. P. S. Bhatia, Akshi Kumar, Rohit Beniwal. **Ontology based framework for detecting ambiguities in software requirements specification**. Published 16 March 2016 Computer Science 2016 3rd International Conference on Computing for Sustainable Global Development.

MUJAHID, A. Miransky. **Ordering interrogative questions for effective requirements engineering: The W6H pattern**. In: Padrões de Requisitos (RePa), 2015 IEEE Fifth International Workshop on, pp. 1-8, 2015.

PRESSMAN, Roger S. **Engenharia de Software**. 6ª ed. Rio de Janeiro: McGraw-Hill, 2006.

RESEARCH GATE. **Construção de um Sistema para a Vigilância Epidemiológica de Traumas na Unidade de Emergência do Hospital das Clínicas de Ribeirão Preto**. Disponível em: https://www.researchgate.net/publication/322697649_Construcao_de_um_Sist

ema_para_a_Vigilancia_Epidemiologica_de_Traumatas_na_Unidade_de_Emergencia_do_Hospital_das_Clinicas_de_Ribeirao_Preto> Acesso em: 15 de jan de 2022.

R. PRIKLADNICKI et al. **The Evolution and Impact of the Research in Distributed Software Development in Brazil**. In: 2011 25th Brazilian Symposium on Software Engineering, 2011, pp. 126-131, doi: 10.1109/SBES.2011.43.

SAMPADA, G. C.; SAKE, Tende Ivo; CHHABRA, Megha. **A Review on Advanced Techniques of Requirement Elicitation and Specification in Software Development Stages**. In: 2020 Sixth International Conference on Parallel, Distributed and Grid Computing (PDGC). IEEE, 2020. p. 215-220.

SPOLETINI, Paola; FERRARI, Alessio. Requirements elicitation: a look at the future through the lenses of the past. In: **2017 IEEE 25th International Requirements Engineering Conference (RE)**. IEEE, 2017. p. 476-477.

SOMMERVILLE, Ian. **Engenharia de Software**. 8ª ed. São Paulo: Pearson Addison-Wesley, 2007.

M. SULTAN, A. MIRANSKY, "Ordering Interrogative Questions for Effective Requirements Engineering: The W6H Standard", Padrões de Requisitos (RePa), 2015 IEEE Fifth International Workshop , 2015.

YATCO, Mei C. **Joint Application Design/Development**. Disponível em: <http://www.umsl.edu/~sauterv/analysis/JAD.html>.

ZAPATA, S., Torres, E., Sevilla, G., Aballay, L., & Reus, M. (2012, October). **Effectiveness of traditional software requirement elicitation techniques applied in distributed software development scenarios**. In 2012 XXXVIII Conferência Latinoamericana em Informatica (CLEI).

APÊNDICE A – QUESTIONÁRIO UTILIZADO PARA AVALIAR O ESTUDO DE CASO NO PROJETO

QUESTÕES	Concordo Plenamente	Concordo Parcialmente	Discordo Plenamente	Concordo Parcialmente	Nenhuma das anteriores
A utilização do framework SRAAF melhoraria a etapa da ER(Conseguir identificar a técnica a ser utilizada de forma mais rápida e correta do que levaria sem utilizar o framework).					
A utilização do framework SRAAF melhoraria a qualidade dos cenários que foram levantados(Acredito ter identificado um conjunto de requisitos mais completo do que identificaria sem usar o método).					
O uso do framework SRAAF ajudaria na melhoria da qualidade do documento de requisitos(SRS).					
O uso do framework SRAAF reduziria a ambiguidade dos requisitos elicitados.					
O framework SRAAF é útil para seleção efetiva da técnica de elicitação para o projeto.					
Considero o framework útil na etapa da Engenharia de Requisitos.					

APÊNDICE B – PERGUNTAS QUALITATIVAS PARA AVALIAÇÃO DA APLICAÇÃO DO FRAMEWORK

PERGUNTAS QUALITATIVAS
Os parâmetros significativos foram suficientes para selecionar a técnica? sentiu falta de algum? Se sim, qual?
As técnicas de elicitação de requisitos apresentadas pelo framework foram suficientes? ou faltou alguma? Se sim, qual?
Teve dificuldade em aplicar a técnica de elicitação selecionada pelo framework?(conhece a técnica ou não).
Quais características do framework SRAAF você considera que auxilia na etapa da engenharia de requisitos? Quais características não auxiliam ou não auxiliam da forma como você gostaria?
Quais foram os principais desafios e/ou dificuldades que impediram que você e sua equipe utilizassem o framework SRAAF no estudo?