UNIVERSIDADE FEDERAL DE PERNAMBUCO

CENTRO DE INFORMÁTICA

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

STHÉFANIE DAL MAGRO

ENGENHARIA DE REQUISITOS EM SISTEMAS CRÍTICOS DE SEGURANÇA: uma abordagem para auxiliar na descoberta de requisitos iniciais de segurança

Recife

2021

STHÉFANIE DAL MAGRO

ENGENHARIA DE REQUISITOS EM SISTEMAS CRÍTICOS DE SEGURANÇA: uma abordagem para auxiliar na descoberta de requisitos iniciais de segurança

Trabalho apresentado ao Programa de Pós-graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco, como requisito parcial para obtenção do grau de Mestre em Ciência da Computação.

Área de concentração: Engenharia de Software e Linguagens de programação

Orientador: Prof. Jaelson Freire Brelaz de Castro

Recife

2021

Catalogação na fonte
Bibliotecária Nataly Soares Leite Moro, CRB15-861

D136e Dal Magro, Sthéfanie
Engenharia de requisitos em sistemas críticos de segurança: uma abordagem para auxiliar na descoberta de requisitos iniciais de segurança / Sthéfanie Dal Magro. – 2021.
156 f.: il., fig., tab.

Orientador: Jaelson Freire Brelaz de Castro.
Dissertação (Mestrado) – Universidade Federal de Pernambuco. CIn, Ciência da Computação, Recife, 2021.
Inclui referências e apêndices.

1. Engenharia de software e linguagens de programação. 2. Engenharia de requisitos. 3. Elicitação de requisitos. 4. Sistemas críticos de segurança. I. Castro, Jaelson Freire Brelaz de (orientador). II. Título

005.1 CDD (23. ed.) UFPE - CCEN 2021 – 168

. **Sthéfanie Dal Magro**

“Engenharia de Requisitos em Sistemas Críticos de Segurança: Uma Abordagem para Auxiliar na Descoberta de Requisitos Iniciais de Segurança”

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação.

Aprovado em: 28/04/2021.

BANCA EXAMINADORA

_____

Profa. Dra. Carla Taciana Lima Lourenço Silva Schuenemann

Centro de Informática/ UFPE

_____

Prof. Dr. Johnny Cardoso Marques

Divisão de Ciência da Computação / ITA

_____

Prof. Dr. Jaelson Freire Brelaz Castro
Centro de Informática/ UFPE

(**Orientador**)

AGRADECIMENTOS

Primeiramente agradeço a Deus por todas as conquistas e bençãos, e por tudo de bom que fazes acontecer em minha vida.

Agradeço aos meus pais, Eliana e Marco Antônio, por serem meus parceiros e terem me dado todo suporte necessário para o desenvolvimento desta dissertação! Sem vocês eu não seria nada.

Agradeço aos amigos feitos durante a pós-graduação, por todo apoio e todo compartilhamento de informações, vocês foram essenciais nesta jornada: Davi, Elisa, Larissa, Moniky, Mozart, Múcio, Rodrigo, Sheilane, vocês estarão sempre em meu coração.

Agradeço ao meu namorado, Rossiny, por toda ajuda e toda compreensão e principalmente por acreditar em mim e me incentivar todos os dias a terminar esta jornada.

Agradeço também a todos que de alguma forma estiveram presentes ao meu lado nessa jornada: Nathally, Brenna, Rosanne, Loredana e Amanda, Viviane e Thayná, amo vocês, meninas!

Meu muitíssimo obrigada a Manoel Alves e Lucas Florêncio por terem me orientado na parte estatística!

E por fim, mas não menos importante, gostaria de agradecer muito ao meu orientador, Jaelson Castro, por todas suas orientações durante o desenvolvimento deste trabalho e por me fazer ver uma luz no fim do túnel, quando eu achava que tudo estava desmoronando. Muito obrigada!

RESUMO

Sistemas Críticos de Segurança (SCSs) são considerados sistemas que caso falhem podem levar à perda de vida, perdas financeiras e danos ao meio ambiente. A Engenharia de Requisitos é essencial no desenvolvimento destes sistemas, tendo em vista que a utilização de requisitos inadequados ou incompreendidos são reconhecidos como a principal causa de acidentes e catástrofes relacionados com a segurança. Portanto, os requisitos iniciais de segurança dos SCSs devem ser cuidadosamente identificados e adequadamente modelados. No entanto, a literatura apresenta poucas técnicas de elicitação e modelagem. Ademais, de acordo com Vilela et al. (2020) é mais econômico corrigir problemas na fase da Engenharia de Requisitos do que na etapa de desenvolvimento do sistema em de requisitos voltadas para o domínio de SCSs. Esta pesquisa propõe o desenvolvimento de uma abordagem para auxiliar na descoberta de requisitos iniciais de segurança no contexto de sistemas críticos de segurança que posteriormente serão modelados através da notação iStar4Safety. Inicialmente, foi realizado um levantamento para aquisição de conhecimento acerca do tema, depois foram identificadas as técnicas de elicitação de requisitos para SCSs existentes na literatura. Em seguida, foi desenvolvida uma nova abordagem para elicitação de requisitos iniciais de segurança, uma ferramenta para dar apoio a abordagem e regras de mapeamento em iStar4Safety. Por último, foi realizada uma avaliação da proposta através de um quasi-experimento. Sendo assim, este trabalho propôs a abordagem Elicit4Safety que visa auxiliar na descoberta e modelagem dos perigos e requisitos iniciais de segurança no contexto dos SCSs, além do desenvolvimento de uma ferramenta que dá suporte a abordagem. De acordo com os resultados encontrados na análise de dados qualitativa e quantitativa, obtidos através aplicação de testes estatísticos e um questionário aplicado aos sujeitos experimentais, é possível observar que a abordagem Elicit4Safety é fácil de entender e de ser utilizada.

Palavras-chaves: engenharia de requisitos; elicitação de requisitos; sistemas críticos de segurança.

ABSTRACT

Safety Critical Systems (SCSs) are considered systems that, if they fail, can lead to loss of life, financial losses, and damage to the environment. Requirements Engineering is essential in the development of these systems, considering that the use of inadequate or misunderstood requirements is recognized as the main cause of accidents and catastrophes related to safety. Therefore, the initial safety requirements of SCSs must be carefully identified and appropriately modeled. However, the literature presents few elicitation and requirements modeling techniques focused on the domain of SCSs. This work aims at the development of an approach to assist in the discovery of initial safety requirements in the context of safety critical systems that will later be modeled using iStar4Safety notation. Initially, a bibliographic survey was carried out to acquire knowledge about the topic, then the techniques for eliciting requirements for SCSs existing in the literature were identified. Then we developed a new approach to elicit initial safety requirements, a tool to support the approach, and the rules for mapping in iStar4Safety. Finally, an evaluation of the proposal was carried out through a quasi-experiment. This work proposed the Elicit4Safety approach that aims to assist in the discovery and modeling of hazards and initial safety requirements in the context of SCSs, in addition to the development of a tool that supports the approach. According to the results found in the qualitative and quantitative data analysis, obtained through the application of statistical tests and a questionnaire applied to experimental subjects, it is possible to observe that the Elicit4Safety approach is easy to understand and to be used.

Keywords: requirements engineering; requirements elicitation; safety critical systems.

LISTA DE FIGURAS

LISTA DE QUADROS

LISTA DE ABREVIATURAS E SIGLAS

BPMN	*Business Process Modeling and Notation*
ER	Engenharia de Requisitos
FTA	*Fault Tree Analisys*
HAZOP	Hazard and Operability Study
HO	*Hazard Ontology*
ICI	*Imperial Chemical Industries*
iStar	iStar Modeling Framework
JAD	*Joint Application Development*
LN	Linguagem Natural
PHA	Preliminary Hazard Analysis
RSL	Revisão Sistemática da Literatura
SCS	*Safety-Critical System* – Sistema Crítico de Segurança
SD	*Strategic Dependency* – Dependência Estratégica
SR	*Strategic Rationale* – Raciocínio Estratégico

SUMÁRIO

1 INTRODUÇÃO

Este capítulo apresenta uma visão geral da dissertação, bem como as principais questões que motivaram a realização deste trabalho e o objetivo da pesquisa. As seções a seguir estão estruturadas da seguinte maneira: contexto, caracterização do problema, objetivos, perguntas de pesquisa e conclusão.

1.1 CONTEXTO

De acordo com Du *et al.* (2014) os Sistemas Críticos de Segurança (SCSs) consistem em um conjunto de hardware, software, processos, dados e pessoas que caso falhem, podem resultar em acidentes que provocam danos ao meio ambiente, perdas financeiras, ferimentos e até a perda de vidas. Os SCSs estão presentes nas mais diversas áreas, tais como: médica, aviação, ferroviária, automotiva, robótica, entre outros. Alguns exemplos desses sistemas são as bombas de infusão de insulina, sistemas de controle de tráfego aéreo, ônibus espaciais, dentre outros.

As atividades e o processo da Engenharia de Requisitos (ER) são essenciais no desenvolvimento de SCSs pois buscam evitar a introdução de defeitos, além de mal-entendidos entre engenheiros e desenvolvedores (LEVESON, 2011). Ademais, de acordo com Vilela *et al.* (2020) requisitos iniciais vagos, ambiguidade na especificação de requisitos e confusão entre métodos e ferramentas afetam severamente a qualidade dos sistemas críticos de segurança.

Sendo assim, podemos afirmar que é de extrema importância que os requisitos sejam adequadamente elicitados, modelados e especificados corretamente buscando evitar atraso na entrega do sistema, falta de confiabilidade no uso do sistema e maior custo no desenvolvimento e posteriormente na manutenção do sistema. De fato, sabemos que o custo de correção dos erros de requisitos é muito menor do que a correção de erros que aparecem nos estágios posteriores do processo de desenvolvimento (SOMMERVILLE, 2011).

Para contribuir com uma melhor especificação de requisitos é recomendado a utilização de linguagens de modelagem, tendo em vista que estas linguagens permitem uma melhor visualização e compreensão dos requisitos (DUARTE, 2018). Ribeiro (2019b) propôs uma extensão da popular notação iStar (YU, 1995), voltada para a modelagem de requisitos iniciais de segurança, denominada iStar4Safety. A

extensão desenvolvida por Ribeiro (2019a) permite a representação gráfica dos requisitos iniciais de segurança, objetivos de segurança dos atores, perigos e causas de perigos.

1.2 CARACTERIZAÇÃO DO PROBLEMA

O desenvolvimento de SCSs merece uma atenção especial, tendo em vista que caso estes sistemas falhem, podem causar grandes tragédias. A Engenharia de Requisitos possui um papel primordial no desenvolvimento desses sistemas, principalmente nas etapas de elicitação e documentação dos requisitos, que são essenciais para garantir o atendimento às necessidades dos *stakeholders*, bem como evitar a introdução de defeitos, perigos e riscos. Ademais, de acordo com Vilela *et al.* (2020) é mais econômico corrigir problemas na fase da Engenharia de Requisitos do que na etapa de desenvolvimento do sistema.

No contexto dos sistemas críticos de segurança, é necessário analisar os possíveis perigos que possam levar a acidentes, bem como definir métodos capazes de mitigar e/ou minimizar os danos ocorridos através destes perigos. Para tanto, é necessário que as atividades da Engenharia de Requisitos caminhem lado a lado com as atividades da área de engenharia de segurança. Sendo assim, é de extrema importância a utilização da ER no contexto de sistemas críticos de segurança, já que uma abordagem bem elaborada de ER pode trazer alguns benefícios para o desenvolvimento de SCSs, como exemplo: atingir as metas de tempo, custo e qualidade (VILELA *et al.,* 2017).

De acordo com Medikonda e Panchumarthy (2009) um requisito de um sistema crítico de segurança pode estar relacionado a diversas funções, tais como: controlar ou influenciar diretamente o funcionamento do hardware crítico de segurança, controlar ou influenciar diretamente os sistemas perigosos, monitorar o estado do sistema com o objetivo de garantir sua segurança, detectar riscos e/ou exibir informações relacionadas à proteção do sistema, lidar ou responder às prioridades de detecção de falhas, desativar ou habilitar o software de processamento de interrupção e computar dados críticos de segurança.

Segundo Yeow e Kia Chiam (2014) requisitos especificados de maneira incompleta e incorreta podem fazer com que os sistemas de software relacionados à segurança não atinjam seus objetivos de segurança. Então, reforçamos que é crucial

garantir a segurança do software, identificando os requisitos de segurança do software adequados durante a atividade de elicitação de requisitos.

Após a realização de um levantamento bibliográfico da literatura verificamos que o processo de elicitação de requisitos para Sistemas Críticos de Segurança é complexo. No entanto, foi observada uma escassez de métodos, técnicas e ferramentas da engenharia de requisitos que sejam específicas para o desenvolvimento destes sistemas.

Diante do cenário apresentado, entende-se como relevante um estudo que contribua para a descoberta de requisitos iniciais de segurança da forma mais completa possível, isento de erros, não ambíguos e que permitam a mitigação dos potenciais perigos dos SCSs, bem como a modelagem e especificação destes requisitos em notações apropriadas como a iStar4Safety (RIBEIRO, 2019b). Para suprir esta necessidade, desenvolvemos uma abordagem denominada **Elicit4Safety** e uma ferramenta de mesmo nome para dar suporte a abordagem, que visa permitir a descoberta e mapeamento de requisitos iniciais de segurança, bem como sua posterior modelagem em iStar4Safety.

1.3 OBJETIVOS

Este trabalho tem como objetivo geral o desenvolvimento de uma abordagem que permita auxiliar a integração da Engenharia de Segurança com a Engenharia de Requisitos, através do processo de levantamento e modelagem de requisitos iniciais de segurança no domínio dos SCSs.

1.3.1 Objetivos Específicos

Para atingir o objetivo geral foram elencados alguns objetivos específicos, apresentados a seguir:

- Realizar um levantamento do estado da arte sobre elicitação de requisitos para SCSs para identificar as técnicas existentes e suas limitações;
- Desenvolver uma abordagem que facilite a elicitação de requisitos iniciais de segurança no domínio dos SCSs;
- Desenvolvimento de uma ferramenta para dar suporte a abordagem;

- Atrelar a abordagem desenvolvida com a técnica de modelagem de requisitos iniciais de segurança denominada iStar4Safety;
- Avaliar a abordagem proposta a partir da realização de um quasi-experimento.

1.4 METODOLOGIA DE PESQUISA

A Figura 1 apresenta a metodologia adotada para realizar esta pesquisa fundamenta-se em quatro abordagens, sendo elas: quanto à natureza, quanto à abordagem do problema, quanto aos objetivos e quanto aos procedimentos técnicos.

Figura 1 - Resumo da metodologia de pesquisa

Fonte: Autora (2021)

Quanto à natureza desta pesquisa, classifica-se como aplicada, pois tem como objetivo gerar conhecimento para aplicação prática e se dirige à solução de problemas específicos (GIL, 2010). Sendo assim, na nossa pesquisa buscaremos solucionar o problema da descoberta e modelagem de requisitos de segurança, visando evitar a introdução de defeitos no sistema.

Referente à abordagem, a pesquisa classifica-se como quantitativa e qualitativa, já que serão utilizados critérios estatísticos para análise das respostas da amostra de pesquisa, bem como critérios qualitativos acerca da abordagem proposta. De acordo com Gil (2010), através da pesquisa quantitativa busca-se traduzir em números, as opiniões e informações com o propósito de classificá-los e analisá-los. Segundo Easterbrook *et al.* 2008) através da análise quantitativa é possível visualizar os benefícios de uma técnica específica. Já a pesquisa qualitativa, segundo Stake (2011), se caracteriza por ser “interpretativa, baseada em

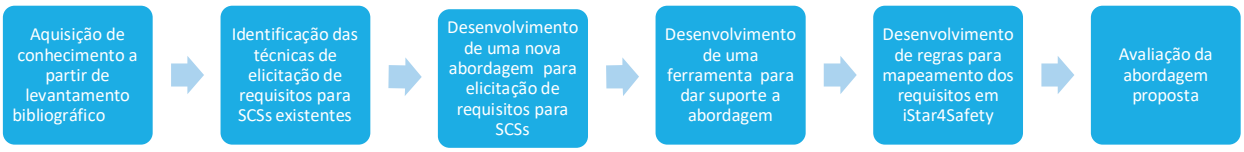
experiências, situacional e humanística".

Quanto aos objetivos, classificou-se esta pesquisa como exploratória, pois possui como objetivo proporcionar maior familiaridade com o problema, buscando torná-lo mais explícito (GIL, 2010).

Quanto ao procedimento técnico, a pesquisa se classifica como experimental, pois determinamos um objeto de estudo, selecionamos as variáveis que seriam capazes de influenciá-lo, definimos as formas de controle e de observação dos efeitos que a variável produz no objeto.

Para atingir nosso objetivo, o trabalho foi conduzido em seis etapas (Figura 2) sendo elas: Aquisição de conhecimento através de um levantamento bibliográfico, identificação das técnicas de elicitação para SCSs já existentes na literatura, desenvolvimento de uma nova abordagem para elicitação de requisitos iniciais de segurança, desenvolvimento de uma ferramenta para dar suporte a abordagem, desenvolvimento de regras para mapeamento dos requisitos em iStar4Safety, ou seja, diretrizes que permitirão que o usuário transforme os requisitos elicitados em modelos na linguagem iStar4Safety. Por fim, será realizada a avaliação da abordagem proposta através de um quasi-experimento.

Figura 2 - Etapas para construção do trabalho



Fonte: Autora (2021)

Para aquisição de conhecimento acerca dos temas propostos, foi realizado um levantamento bibliográfico através de bibliotecas digitais, sendo elas: ACM Digital Library, IEEE Xplore Digital Library, ScienceDirect e Springer, buscando encontrar artigos e livros relevantes para a construção da dissertação.

A partir das informações adquiridas no levantamento bibliográfico, foi possível investigar quais técnicas de descoberta de requisitos no contexto de SCSs existem na literatura. Em seguida, foi elaborada uma nova abordagem para elicitação, com o objetivo de atrelar o levantamento de requisitos iniciais de segurança com a modelagem deles. Para dar suporte a abordagem criada, foi desenvolvida uma ferramenta e regras para mapeamento dos requisitos para construção de modelos

na linguagem iStar4Safety.

Os procedimentos experimentais permitem o isolamento de fatos e parâmetros, permitindo a comparação de um fenômeno em duas ou mais amostras similares. Desta forma, utilizamos o quasi-experimento para comparar as modelagens em iStar4Safety do grupo experimental ao do grupo de controle, onde apenas o grupo experimental fez uso da abordagem Elicit4Safety.

1.5 ESTRUTURA DA DISSERTAÇÃO

Essa dissertação encontra-se dividida em 05 capítulos e 04 apêndices:

- O **capítulo 1** apresenta a introdução do trabalho, bem como seu contexto, objetivos, metodologia de pesquisa e estrutura da dissertação.
- O **capítulo 2** apresenta o referencial teórico com os principais conceitos das áreas envolvidas nesta pesquisa, bem como um relato dos trabalhos relacionados. Especificamente, são abordados os seguintes temas: Sistemas Críticos de Segurança, Segurança, Engenharia de Requisitos, iStar4Safety, Trabalhos Relacionados.
- O **capítulo 3** apresenta a abordagem Elicit4Safety, seu processo de construção e a ferramenta desenvolvida para auxiliar a utilização da abordagem.
- O **capítulo 4** define como foi realizada a avaliação da abordagem Elicit4Safety, através de um quasi-experimento. Também é apresentada uma avaliação qualitativa da abordagem, com os dados adquiridos pelos participantes do grupo experimental, após a utilização da ferramenta.
- O **capítulo 5** apresenta a conclusão desta dissertação, discutindo os resultados obtidos, expondo suas limitações encontradas e propondo alguns trabalhos para o futuro.
- O **apêndice A** apresenta os materiais utilizados na preparação do quasi-experimento.
- O **apêndice B** apresenta o mapeamento das informações de segurança do robô Miras através da utilização do Elicit4Safety e iStar4Safety.
- O **apêndice C** apresenta os dados brutos obtidos com a realização do quasi-experimento.

2 FUNDAMENTAÇÃO TEÓRICA E TRABALHOS RELACIONADOS

Este capítulo apresenta os conceitos necessários para o entendimento desta dissertação. Serão abordados os seguintes temas: Segurança, Sistemas Críticos de Segurança, Engenharia de Requisitos, Elicitação de requisitos, Técnicas de Elicitação de Requisitos, Documentação de Requisitos e a notação para modelagem de requisitos iniciais de segurança denominada iStar4Safety. Concluímos com um relato dos trabalhos relacionados que nos auxiliaram na aquisição do conhecimento necessário para o desenvolvimento do nosso projeto de pesquisa.

2.1 SISTEMA CRÍTICO DE SEGURANÇA (SCS)

Sistema Crítico de Segurança (SCS), do inglês *Safety Critical System* é aquele sistema cuja falha pode resultar em perda de vida, danos significativos à propriedade ou danos ao meio ambiente, podendo ser encontrados em diversas áreas de aplicação, tais como: dispositivos médicos, controle de voo de aeronaves, armas, sistemas nucleares, sistemas robóticos (KNIGHT, 2002). De acordo com Lutz (2000), a tendência é que com os avanços tecnológicos e o aumento de mercados consumidores, sejam produzidos sistemas mais críticos à segurança.

Segundo Grant (2016), a principal característica de um sistema crítico de segurança é o alto grau de complexidade, e, geralmente, são sistemas em tempo real que interagem com o ambiente e os usuários de diversas maneiras. Knight (2002) afirma que a falha de um sistema pode levar a consequências que são consideradas inaceitáveis. O processo de levantamento de requisitos para Sistemas Críticos de Segurança é bem complexo, uma vez que é necessário capturar os comportamentos de todos os subsistemas envolvidos, bem como integrar diversas restrições (BROOMFIELD AND CHUNG, 1997).

Para Lutz (2000) o software é essencial para que a maioria dos sistemas críticos de segurança alcancem seus objetivos. No entanto, de acordo com Vilela *et al.* (2018a), o software está se tornando uma fonte e riscos e contribuindo com catástrofes ligadas à segurança, tendo em vista que além de controlar um número crescente de funções tradicionais e inovadoras, o software também está lidando com funções que antes eram controladas por seres humanos, de modo que vem se tornando uma importante fonte de riscos e perigos, uma vez que a transmissão de

instruções erradas ao hardware pode levar à acidentes e lesão de pessoas. Lutz (2000) afirma que o software pode tanto contribuir para a segurança de um sistema, como comprometê-lo, colocando-o em um estado perigoso. Neste ponto de vista, o desenvolvimento de sistemas críticos com a confiabilidade adequada exige avanços em áreas como arquitetura, verificação e processo (KNIGHT, 2002).

Pode-se afirmar que a confiabilidade dos SCSs depende bastante que as preocupações com segurança ocorram no início do processo de desenvolvimento do sistema, com a utilização da engenharia de requisitos. Kumar *et al. (2010)* relatam que a segurança de software deve ser aplicada em um sistema até o dia em que ele se aposente. Portanto, é importante que exista uma documentação adequada ao projeto, pois permitirá um melhor entendimento do sistema e facilitará o processo de atualizações do sistema, caso seja necessário (GRANT, 2016). Sendo assim, pode-se afirmar que é de extrema importância um correto e preciso levantamento dos requisitos de segurança, bem como sua modelagem e documentação.

Kumar *et al. (2010)* lista alguns pontos relacionados à segurança do software, sendo eles:

- Documentação dos planos de segurança, das decisões, processos e resultados;
- Integrar a segurança no ciclo de desenvolvimento do software;
- Análise do software, do sistema e das interfaces desde o início até à finalização;
- Rastreamento de requisitos de segurança do software em todas as fases de desenvolvimento;
- Controlar a configuração do software; e
- Relatar e resolver os problemas.

Como visto anteriormente, é extremamente importante que os requisitos sejam elicitados e especificados de maneira precisa e completa, pois caso especificados de maneira incorreta ou incompleta podem levar à danos e acidentes. De acordo com Firesmith (2004) a especificação é incompleta quando não menciona como o sistema deve evitar ou eliminar os riscos e como o sistema deve se comportar quando ocorrem riscos ou incidentes de segurança, bem como quando não definem o comportamento que o sistema deve possuir em todas as

combinações possíveis de estados, ou até mesmo como o sistema deve lidar com circunstâncias excepcionais.

Diante do exposto, mostra-se essencial a preocupação com segurança desde o início do desenvolvimento de sistemas críticos, pois só assim serão alcançados altos níveis de segurança, considerando que a adição de componentes de proteção e complexidades adicionais após o desenvolvimento do sistema não resolve o problema (VILELA *et al.*, 2018a).

2.2 SEGURANÇA

Como nossa pesquisa está voltada para a área de segurança, do inglês *safety,* se faz necessário definir seus principais conceitos e técnicas de análise. Leveson (1995) classifica segurança, como a ausência de acidentes ou perdas. É essencial que exista o gerenciamento de perigos, ou seja, a identificação, avaliação, eliminação e controle por meio de procedimentos de análise e gerenciamento (LEVESON, 1995).

2.2.1 Conceitos de segurança

A literatura apresenta algumas definições relacionadas à segurança, sendo elas:

- Risco: É a combinação da probabilidade de um evento ou falha anormal e a consequência desse evento ou falha nos componentes, operadores, usuários ou ambiente de um sistema (IEEE, 2000).
- Perigo: O IEEE (2000) apresenta duas definições para perigo: (1) uma propriedade ou condição intrínseca que possui o potencial de causar dano, (2) uma condição ou potencial existente que pode resultar em um acidente. Já Leveson (1995) define perigo como um estado ou conjunto de condições de um sistema (ou um objeto) que, juntamente com outras condições no ambiente do sistema (ou objeto) levará inevitavelmente a um acidente (evento de perda).
- Acidente: É um evento não planejado ou uma série de eventos que resultam em morte, lesão, danos ao equipamento ou propriedade ou à perda de propriedade, ou danos ao meio ambiente (IEEE, 2000).

- Confiabilidade: É a probabilidade de que um equipamento ou componente execute sua função pretendida de maneira satisfatória por um tempo prescrito e em condições previamente estipuladas (LEVESON, 1995).
- Falha: É a incapacidade do sistema ou componente de executar sua função pretendida por um tempo especificado em condições ambientais específicas (LEVESON, 1995).
- Erro: é uma falha do projeto ou o desvio do estado pretendido (LEVESON, 1995).
- Causa do Perigo: De acordo com Ribeiro (2019b) uma causa do perigo é refere-se a uma condição é suficiente para que o perigo relacionado a ela ocorra.
- Condição ambiental: De acordo com Ribeiro (2019b), as condições ambientais referem-se aos componentes e as suas propriedades que apesar de não fazerem parte propriamente do sistema, podem afetar o comportamento do sistema.
- Requisitos funcionais de segurança: De acordo com Ribeiro (2019b) "[...] são os requisitos usados para mitigar ou prevenir os efeitos de falhas identificadas na análise de segurança".
- Estratégias de segurança: De acordo com Ribeiro (2019b) tratam-se de "ações que visam mitigar as consequências de um possível acidente."
- Recursos de segurança: De acordo com Vilela *et al.* (2017b apud Ribeiro, 2019b) os recursos são os ativos necessários para o correto funcionamento de requisitos críticos, portanto são modelados como especializações de recursos, indicando assim a sua criticidade em relação a outros recursos.

2.2.2 Análise preliminar de segurança

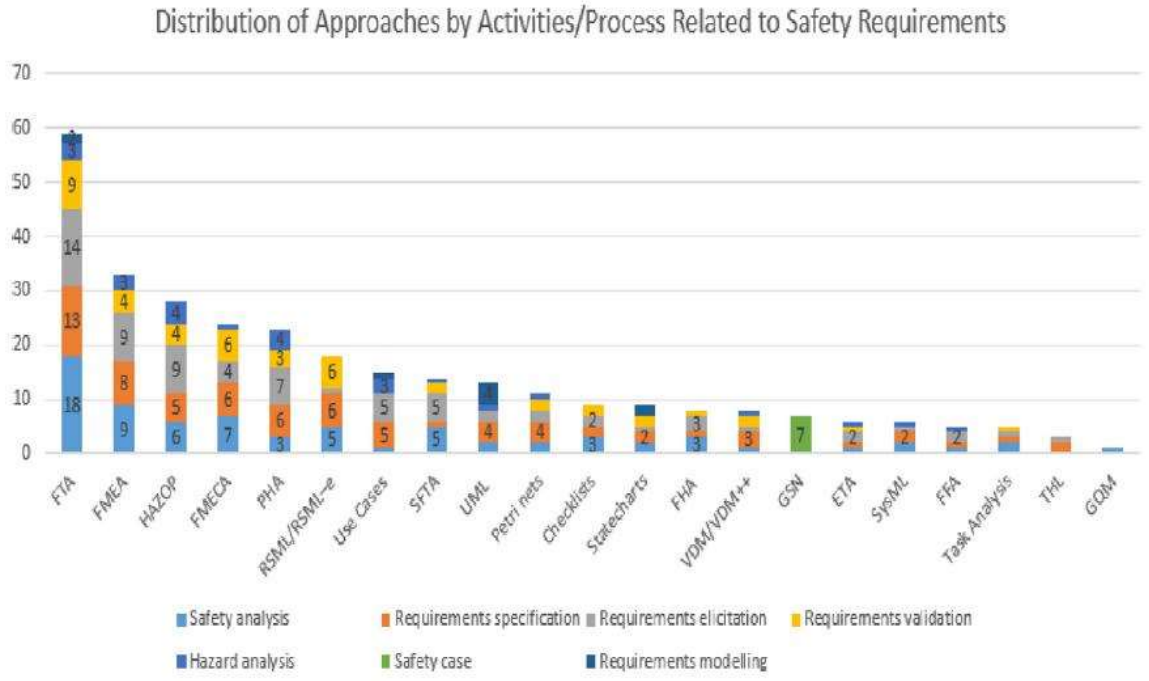
A Análise Preliminar de Segurança visa identificar os perigos e riscos do sistema, sendo um pré-requisito importante para auxiliar na identificação dos requisitos de segurança. Esta análise permite a aquisição de uma visão geral dos pontos necessários para garantir a segurança do sistema.

Firesmith (2004) defende que esta etapa permite a categorização da gravidade de acidentes e perigos, bem como a definição da probabilidade em que

esses acidentes e perigos podem ocorrer. Para realização da Análise Preliminar de Segurança podem ser utilizadas as mais diversas técnicas, no entanto, Vilela *et al.* (2017) identificaram através de uma revisão sistemática que as técnicas mais utilizadas são: *Preliminary Fault Tree Analysis* (FTA), *Preliminary Hazard Analysis (PHA)* e *Hazard and Operability Studies* (HAZOP).

Martins e Gorschek (2016) elaboraram uma revisão sistemática da literatura com o objetivo de investigar quais abordagens estão sendo propostas para elicitar, modelar, especificar ou validar requisitos de segurança no domínio de Sistemas Críticos de Segurança. Eles identificaram 165 estudos que tratavam de diversos domínios de aplicação, dentre eles aeronáutico, médico, aeroespacial, ferroviário. Os resultados obtidos por Martins e Gorschek (2016) apresentam a distribuição das abordagens de segurança nas atividades/processos relacionados aos requisitos de segurança, conforme mostra a Figura 3. A etapa da elicitação de requisitos está representada pela cor cinza, sendo assim, observa-se que abordagem mais utilizada para a descoberta de requisitos é o FTA, seguido pelo FMEA, HAZOP e PHA. A seguir, iremos apresentar as técnicas FTA, HAZOP e PHA.

Figura 3 - Distribuição das Abordagens por Atividades/Processos relacionados aos requisitos de segurança



Fonte: Martins e Gorsheck (2016)

2.2.2.1 *Fault Tree Analysis* (FTA)

O FTA foi desenvolvido em 1961 por H.A. Watson nos Laboratórios Bell e é uma técnica amplamente utilizada em vários contextos. Esta técnica é representada através de uma árvore, com o objetivo de analisar as causas de perigos. O evento do topo da árvore deve ser previsto e identificado primeiramente através de outras técnicas, como exemplo, o HAZOP (LEVESON, 1995). De acordo com Broomfield e Chung (1997) o FTA baseia-se na seleção de um evento principal e na avaliação da combinação de falhas e condições que podem fazer com que o evento principal ocorra.

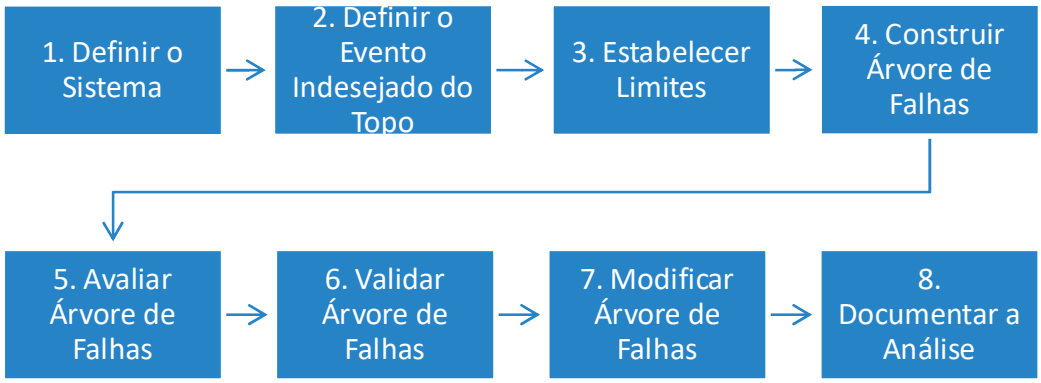
Ericson (2005) afirma que os resultados do FTA podem auxiliar os stakeholders a realizar as seguintes atividades: (1) Verificar a conformidade do projeto com os requisitos de segurança estabelecidos, (2) Identificar as deficiências de segurança do projeto que se desenvolveram apesar dos requisitos existentes (3) Identificar as falhas de modo comum, (4) Estabelecer medidas preventivas para mitigar ou eliminar deficiências de segurança de projeto, (5) Avaliar a adequação das medidas preventivas estabelecidas e (6) Estabelecer ou modificar requisitos de segurança adequados para a próxima fase do projeto. Esta última atividade auxilia na elicitação dos requisitos iniciais de segurança.

Ainda de acordo com Ericson (2005) o processo para realização de uma análise FTA (Figura 4) em um sistema consiste em 8 passos:

- Definir o Sistema: Consiste na compreensão do projeto e da operação do sistema.
- Definir o Evento Indesejado do Topo: Consiste na definição descritiva do problema e estabelecimento do evento indesejado para a análise.
- Estabelecer Limites: Consiste na definição de regras básicas de análise e de limites. Deve-se avaliar o problema e registrar todas as regras básicas.
- Construir Árvore de Falhas: Consiste no processo de construção, regras e lógicas para criar o modelo de árvore de falhas no sistema.

- Avaliar Árvore de Falhas: Para avaliar a árvore de falhas é necessário gerar um conjunto de cortes e probabilidade, buscando identificar os elos fracos e os problemas de segurança do projeto.
- Validar Árvore de Falhas: Verificar se o modelo da árvore de falhas está correto, completo, preciso e se reflete o projeto do sistema.
- Modificar Árvore de Falhas: Consiste na modificação da árvore de falhas conforme necessidade identificada na fase de validação.
- Documentar a Análise: Consiste na documentação de toda a análise com dados de apoio.

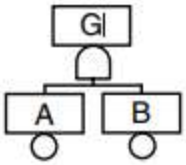
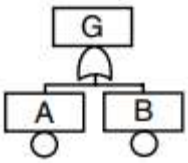
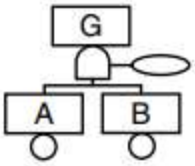
Figura 4 - Processo para realização do FTA em sistemas

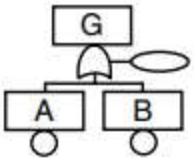
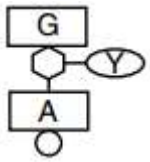


Fonte: Baseado em Ericson (2005)

Para construção do FTA é necessário identificar o perigo principal que será o topo da árvore e a partir de então realizar o refinamento deste perigo, através da utilização da lógica booleana. O Quadro 1 apresenta os principais símbolos para construção da árvore de falhas.

Quadro 1 - Símbolos do FTA

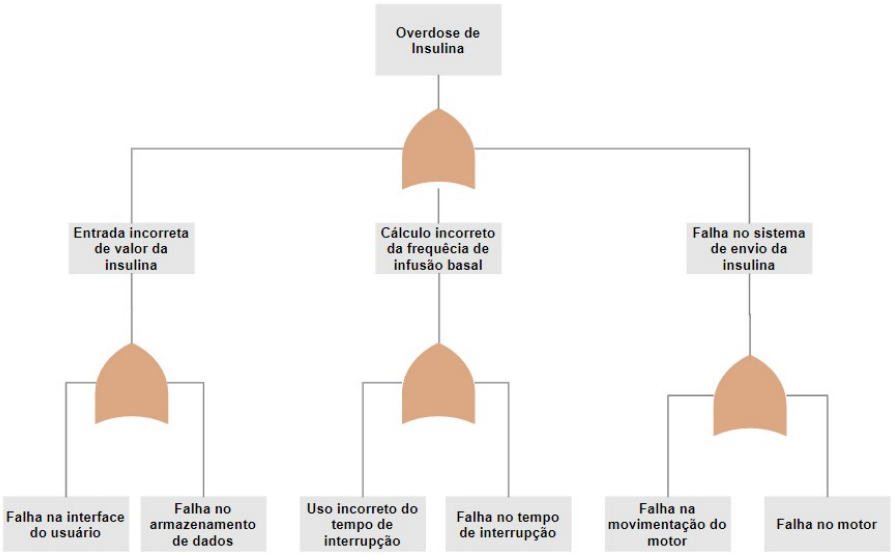
Símbolos	Tipo	Descrição
	Caixa de texto dos nós	Contém o texto dos nós da árvore de falhas.
	Falha primária	Representa uma falha básica de componente.
	Falha secundária	Representa uma falha induzida externamente ou um modo de falha que, se desejado, pode ser explorado em mais detalhes
	Evento Normal	Representa um evento que ocorre como parte da operação normal do sistema.
	Portão *AND*	A saída apenas ocorre se todas as entradas ocorrem.
	Portão *OR*	A saída apenas se pelo menos uma das entradas ocorre.
 x	Portão *Priority AND*	A saída ocorre apenas se todas as entradas ocorrerem, e A deve ocorrer antes de B. A declaração de prioridade está contida no símbolo de condição.

	Portão *Exclusive OR*	A saída ocorre se uma das entradas ocorrer, mas não ambas. A declaração de exclusividade está contida no símbolo de condição.
	Portão *Inhibit*	A saída ocorre apenas se o evento de entrada ocorrer e a condição anexada for satisfeita.

Fonte: Ericson (2005)

A Figura 5 apresenta um exemplo de FTA do perigo “Overdose de Insulina”, de uma bomba de infusão de insulina:

Figura 5 - Árvore de Falhas do perigo “Overdose de Insulina” de uma Bomba de Infusão de Insulina



Fonte: Martins e De Oliveira (2014)

Conforme a Figura 5, existem três possibilidades para que o evento principal “Overdose de Insulina” ocorra, sendo elas:

(1) Entrada incorreta de valor da insulina

Para que este evento ocorra houve uma falha na interface do usuário ou uma falha no armazenamento dos dados.

OU

(2) Cálculo incorreto de frequência de infusão basal

Para que este evento ocorra houve o uso incorreto do tempo de interrupção ou falha no tempo de interrupção.

OU

(3) Falha no Sistema de Envio da Insulina

Para que este evento ocorra houve uma falha na movimentação do motor ou uma falha no motor.

2.2.2.2 *Hazard and operability analysis* (HAZOP)

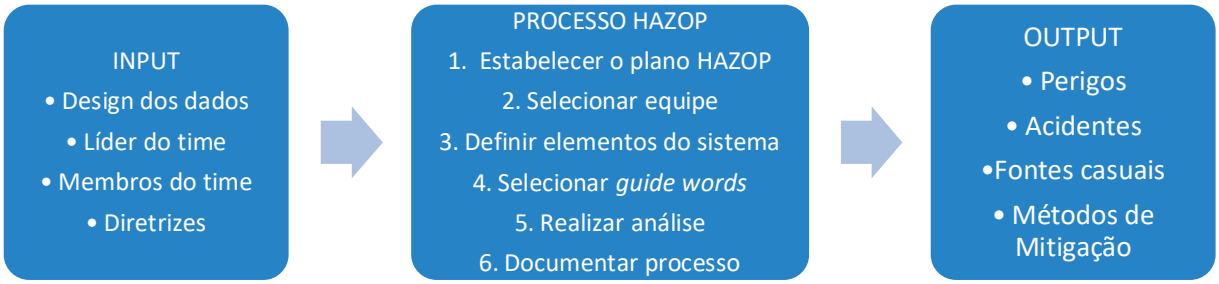
O HAZOP foi desenvolvido nos anos 60 pela Imperial Chemical Industries (ICI), e, de acordo com Leveson (1995) a técnica é baseada em um modelo de teoria de sistemas de acidentes, que assume a causa dos acidentes através de desvios do projeto ou intenções de operação, para tanto, a técnica incentiva o pensamento criativo sobre todas as formas possíveis em que perigos ou problemas operacionais podem surgir.

O HAZOP deve ser executado sistematicamente, considerando cada unidade de processo na planta e cada risco de maneira individual, buscando reduzir a chance de que algo seja esquecido (LEVESON, 1995).

De acordo com Ericson (2005) a análise HAZOP envolve a obtenção de uma descrição completa do sistema e o questionamento de cada componente dele, objetivando identificar os desvios de comportamento que possam surgir. A partir da identificação dos desvios é necessário a realização de uma avaliação dos efeitos negativos que esses desvios possam trazer. A análise HAZOP deve ser realizada com uma equipe multidisciplinar, permitindo que todos os aspectos do sistema sejam observados e questionados.

A Figura 6 apresenta o processo para realização da análise HAZOP, incluindo suas entradas, etapas e saídas.

Figura 6 - Processo para realização do HAZOP



Fonte: Ericson (2005)

Como mencionado anteriormente, a base da técnica HAZOP consiste em um conjunto palavras guia (*guide words)*, ou seja, uma abordagem exploratória para identificação de perigos (BROOMFIELD E CHUNG, 1997).

O Quadro 2 apresenta as principais *guide words* do HAZOP.

Quadro 2 - Palavras Chaves do HAZOP

Guide word / Palavra Chave	**Significado**
No / Não	Quando a intenção do projeto não ocorre ou não se alcança o aspecto operacional.
Less / Menos	Quando ocorre uma diminuição quantitativa na intenção do projeto
More / Mais	Quando ocorre um aumento quantitativo na intenção do projeto
Reverse / Reverso	Quando ocorre o oposto da intenção do projeto
Also / Também	Quando a intenção do projeto é totalmente cumprida, mas, além disso, ocorre alguma outra atividade relacionada
Part of / Parte de	Quando apenas parte da etapa é realizada
Fails / Falha	Falha para operar ou realizar a intenção proposta.
Before/ After Antes /Depois	Ocorre quando a etapa (ou alguma parte dela) é efetuada fora da sequência
Faster/Slower Mais rápido / Mais Lento	Quando a etapa foi realizada / não realizada no momento certo.

Fonte: Adaptado de Ericson (2005)

As *guide words* ajudam a direcionar e estimular o processo criativo na identificação de possíveis desvios de projeto. A partir da identificação desses possíveis desvios de projeto, é possível elicitar os requisitos de segurança capazes de mitigar este perigo. O Quadro 3 mostra um exemplo de aplicação do HAZOP na função de “Movimento com detecção de obstáculos” do Robô MIRAS (Anexo A.1)

Quadro 3 - Aplicação do HAZOP no Robô MIRAS

Elemento	**Guideword**	**Derivação**	**Possíveis Causas**	**Requisito de Segurança**
Obstáculo detectado no caminho	Não	O obstáculo não é detectado	Erro na detecção de obstáculos (HW / SW)	Nenhuma parte saliente deve existir no robô
	Parte de	O obstáculo é detectado, mas na posição errada	Erro na detecção de obstáculos (HW / SW)	Nenhuma parte saliente deve existir no robô

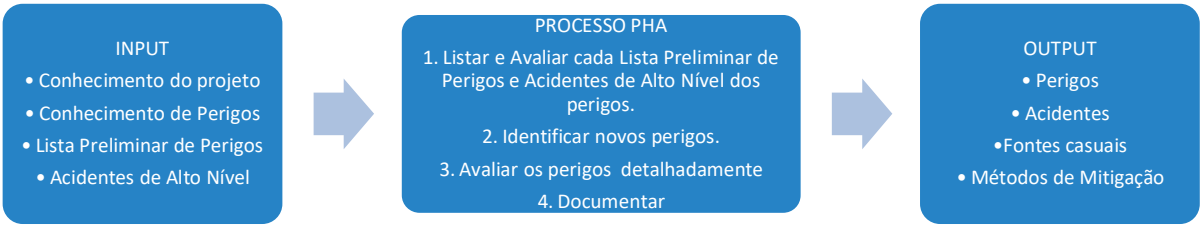
Fonte: Miras (2009)

2.2.2.3 *Preliminary Hazard Analysis* (PHA)

Broomfield e Chung (1997) afirmam que a construção de requisitos de segurança depende da avaliação do ambiente que o sistema está envolvido, buscando identificar potenciais perigos.

De acordo com Ericson (2005) a técnica de Análise Preliminar de Perigos, do inglês *Preliminary Hazard Analysis (PHA)*, consiste em uma ferramenta de análise de segurança para identificar perigos, seus fatores, efeitos e níveis de risco e mitigação deles. O PHA fornece um método eficaz (Figura 7) para a descoberta e comparação de perigos do sistema, auxiliando na descoberta dos requisitos iniciais de segurança do sistema. No entanto, para realizar a PHA são necessárias algumas entradas: conhecimento do projeto, conhecimento dos perigos, uma lista preliminar dos perigos e uma lista com os acidentes de alto nível. Ademais, o PHA é uma técnica rápida e fácil de ser aplicada.

Figura 7 - Processo para realização do PHA



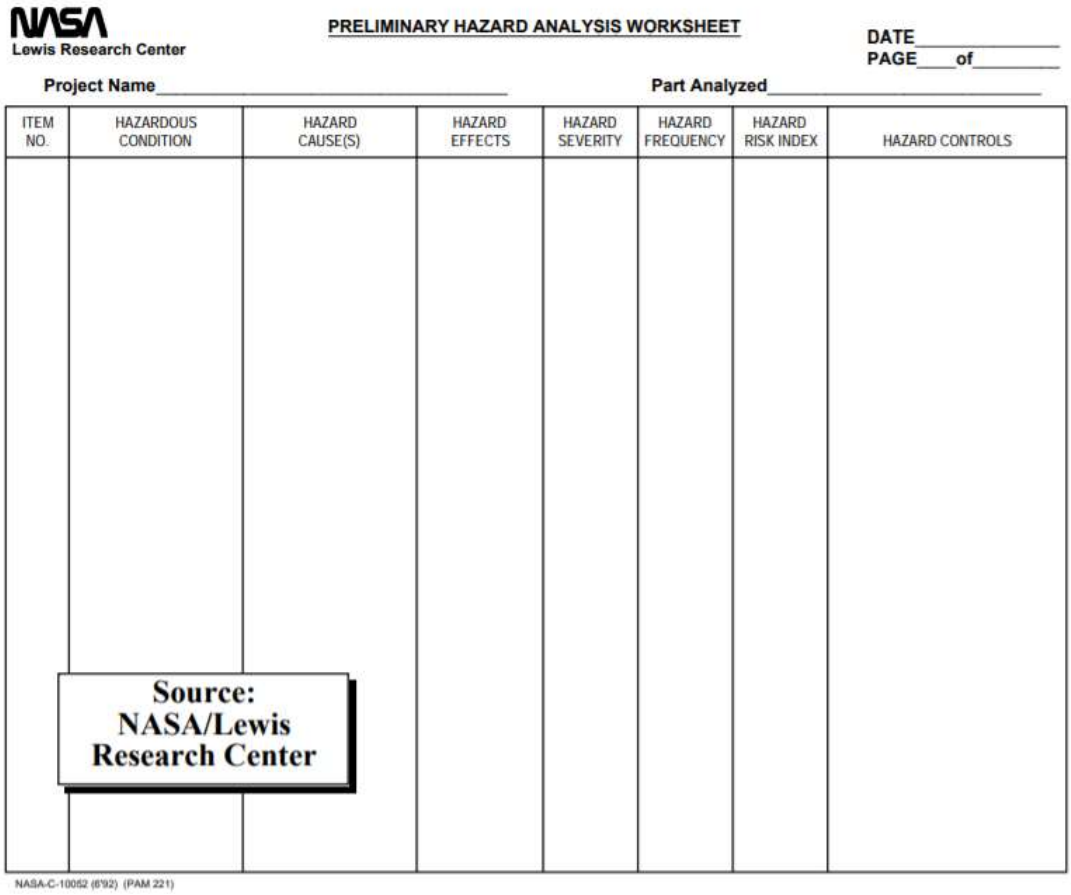
Fonte: Adaptado de Ericson (2005)

Para a correta aplicação da PHA, é desejável que se utilize uma planilha especializada, pois esta planilha permitirá a descoberta e descrição dos perigos, bem como suas estratégias de mitigação, servindo como artefato para a elicitação de requisitos de segurança.

De acordo com Ericson (2005), uma planilha de PHA deve conter os seguintes itens: Perigos do sistema, os efeitos dos perigos, suas causas, avaliação de risco de falha, recomendações para eliminar ou mitigar os perigos. Na literatura existem diversas tabelas para o preenchimento da PHA, entre elas a da NASA Lewis Research Center, representada na Figura 8. A escolha da apresentação desta planilha se deu por conta da similaridade de conceitos com o iStar4Safety.

A abordagem descrita nesta dissertação poderá ser alinhada com técnica PHA, já que apresenta conceitos parecidos com os utilizados em iStar4Safety, sendo eles: Condição Perigosa, Causa do Perigo, Efeito do Perigo, Severidade do Perigo.

Figura 8 - Tabela PHA Nasa Lewis Research Center


NASA
Lewis Research Center

PRELIMINARY HAZARD ANALYSIS WORKSHEET

DATE_____
PAGE____of_____

Project Name_____ Part Analyzed_____

ITEM NO.	HAZARDOUS CONDITION	HAZARD CAUSE(S)	HAZARD EFFECTS	HAZARD SEVERITY	HAZARD FREQUENCY	HAZARD RISK INDEX	HAZARD CONTROLS

Source: NASA/Lewis Research Center

NASA-C-10052 (6'92) (PAM 221)

Fonte: Nasa Lewis Research Center[1]

2.3 ENGENHARIA DE REQUISITOS

Requisitos são definidos durante os primeiros estágios do desenvolvimento do sistema, eles descrevem as funcionalidades do sistema, os serviços oferecidos, informações de domínio de aplicativo, restrições de operação do sistema ou especificações de um sistema, podendo variar de acordo com a necessidade dos clientes (KOTONYA E SOMMERVILLE, 1998). Os requisitos podem ser classificados como: (1) funcionais e (2) não funcionais, de modo que o primeiro é responsável por declarar os serviços que deverão ser realizados pelo sistema, além de como o sistema reagirá a entradas especificas e seu comportamento em algumas situações,

[1] Disponível em: https://hsseworld.com/wp-content/uploads/2017/05/PHA.pdf

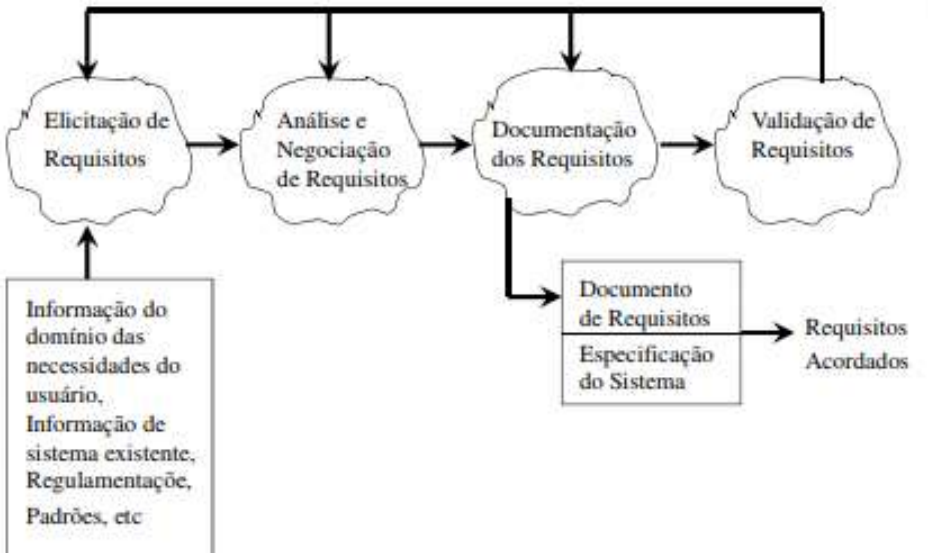
já o segundo, apresenta as restrições aos serviços/funções oferecidas pelo sistema (SOMMERVILLE, 2011).

O uso do termo ‘engenharia’ implica na utilização das técnicas capazes de certificar que os requisitos do sistema possuem qualidade, ou seja, se são completos, consistentes, relevantes, se não possuem ambiguidade (KOTONYA E SOMMERVILE, 1998). Portanto, a engenharia de requisitos tem como objetivo descobrir, analisar, documentar e validar os requisitos para o desenvolvimento de sistemas (SOMMERVILLE, 2011).

O erro na especificação de requisitos pode trazer alguns problemas, como exemplo: atraso na entrega do sistema, falta de confiabilidade no uso do sistema e maior custo no desenvolvimento e posteriormente na manutenção, além disto, o custo de correção dos erros de requisitos são muito maiores do que a correção de erros que aparecem nos estágios posteriores do processo de desenvolvimento, tendo em vista que a correção de problemas de requisitos pode exigir um retrabalho nas equipes de projeto, implementação e teste do sistema (KOTONYA E SOMMERVILLE, 1998).

Na literatura podem ser encontradas diversas classificações para o processo de Engenharia de Requisitos. No entanto, neste trabalho, iremos utilizar a classificação de Kotonya e Sommerville (1998) (Figura 9) que inclui quatro atividades de alto nível, sendo elas: elicitação de requisitos, análise e negociação de requisitos, documentação de requisitos e validação de requisitos. É importante ressaltar que não possui limites distintos entre as atividades e em prática estas atividades são intercaladas, possuindo bastante interação entre as atividades (KOTONYA E SOMMERVILLE, 1998).

Figura 9 - Atividades do processo de engenharia de requisitos



Fonte: Kotonya e Sommerville (1998)

As etapas podem ser classificadas da seguinte maneira: Na primeira etapa, de elicitação de requisitos, os requisitos são levantados através de uma consulta com os *stakeholders*, ou seja, as partes interessadas, bem como outras fontes de informação. A segunda etapa, de análise e negociação é responsável pela análise dos requisitos, em caso de conflitos, os mesmos deverão ser resolvidos por meio de uma negociação. A terceira fase, chamada de documentação dos requisitos é responsável pela produção de um documento que contenha todos os requisitos especificados para que na próxima etapa, de validação, seja checado a consistência do documento produzido, buscando verificar se ele condiz com aquilo que os *stakeholders* solicitaram (KOTONYA, 1998). O foco dos nossos trabalhos será na elicitação dos requisitos de segurança e em sua posterior modelagem.

2.3.1 Elicitação de Requisitos

Esta etapa, consiste na descoberta dos requisitos para o desenvolvimento do sistema e permite a externalização do conhecimento entre os *stakeholders,* que devem buscar a troca de informações acerca do desenvolvimento do sistema, bem como suas funcionalidades e restrições. No entanto, de acordo com Kausar *et al.* (2010) os *stakeholders* possuem diferentes percepções acerca do sistema, acarretando uma grande variedade no processo de engenharia de requisitos. Pohl (2010) classifica três objetivos da elicitação de requisitos, sendo eles:

(1) Identificar fontes de requisitos relevantes;

(2) Elicitar requisitos existentes das fontes identificadas;

(3) Desenvolver requisitos novos e inovadores.

Segundo Kotonya e Sommerville (1998), o processo de elicitação de requisitos pode ser dividido em quatro atividades a serem desenvolvidas (Figura 10).

(1) **Entendimento do domínio da aplicação:** Entender o domínio de aplicação significa compreender, de modo geral, o contexto de onde o sistema será aplicado.

(2) **Entendimento do problema:** Esta atividade busca compreender quais os detalhes específicos referente ao problema do cliente e onde o sistema será aplicado.

(3) **Entendimento do negócio:** Esta atividade tem como objetivo o entendimento da interação e contribuição do sistema com os objetivos do negócio.

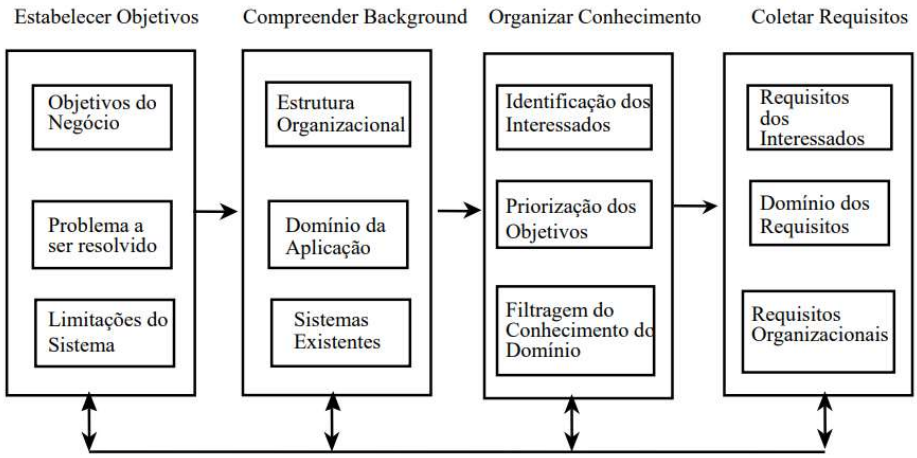
(4) **Entendimento das necessidades e restrições dos *stakeholders*:** Esta atividade busca entender os processos de trabalho que o sistema pretende apoiar e a função de sistemas existentes neste processo de trabalho.

Figura 10 - Componentes da elicitação de requisitos

Fonte: Kotonya e Sommerville (1998)

De acordo com Kotonya e Sommerville (1998) o processo de elicitação de requisitos é dividido em quatro etapas, conforme Figura 11.

Figura 11 - Processo de elicitação de requisitos



Fonte: Kotonya e Sommerville (1998)

A primeira etapa busca definir objetivos, de modo que devem ser estabelecidos os objetivos organizacionais, abrangendo os objetivos gerais do negócio, uma descrição do problema a ser tratado, a necessidade e as limitações do sistema. A segunda etapa, consiste em compreender o background, ou seja, deverá compreender quais informações referentes ao background do sistema, as informações referentes ao local de instalação do sistema, o domínio de aplicação do sistema e informação acerca de outros sistemas existentes. Já na terceira etapa, denominada de organização do conhecimento, espera-se que o conhecimento adquirido nas etapas anteriores seja organizado e colocado em ordem. A última etapa, consiste em coleta os requisitos do *stakeholders*, ou seja, as partes interessadas deverão ser consultadas para descobrir os requisitos que deverão ser implementados no sistema (KOTONYA E SOMMERVILLE, 1998).

De acordo com Kotonya e Sommerville(1998) o processo de elicitação de requisitos não é tão simples quanto aparenta, tendo em vista que os clientes raramente possuem uma visualização clara dos requisitos, além de que diferentes pessoas na organização podem possuir requisitos conflitantes.

Tsumaki e Tamai (2006) apresentam uma lista de problemas que ocorrem na etapa de elicitação de requisitos, conforme Quadro 4.

Quadro 4 - Problemas com requisitos

Requisitos Incompletos	**Requisitos Inconsistentes**
Entendimento das necessidades incompletos	Intenções não sólidas dos solicitantes
Conhecimento do domínio incompleto	Diferentes pontos de vistas de diferentes usuários
Negligenciar suposições tácitas	**Requisitos não corrigidos**
Pouca colaboração dos usuários	Requisitos flutuantes
Requisitos Incorretos	Aceitação contínua de requisitos adicionais
Limite do sistema mal definido	**Requisitos excessivos**
Incompreensão do propósito do sistema	Fonte de informação desorganizada
Requisitos Ambíguos	Muitos solicitantes
Termos sinônimos e homônimos	Consideração de design desnecessária
Termos não testáveis	

Fonte: Tsumaki & Tamai (2006)

De acordo com Broomfield e Chung (1997) no âmbito do desenvolvimento de sistemas críticos, a fase de elicitação de requisitos é bem complexa, tendo em vista que é necessário capturar o comportamento pretendido do sistema e integrar com diversas restrições, envolvendo diferentes profissionais, sobre o comportamento pretendido. Sendo assim, para uma correta descoberta dos requisitos é recomendado que haja uma integração com a engenharia de segurança, permitindo que todos os participantes possam analisar os riscos potenciais do sistema (BROOMFIELD, CHUNG, 1997).

2.3.1.1 Técnicas de Elicitação de Requisitos

As técnicas de elicitação permitem a identificação das informações acerca do domínio da aplicação e do problema específico que necessita ser resolvido. Como a literatura apresenta diversas de técnicas de elicitação, o profissional da área de engenharia de requisitos deverá escolher quais técnicas utilizar para a coleta de informações acerca dos requisitos que os *stakeholders* almejam. No entanto, escolher a técnica ou a combinação de técnicas de elicitação mais adequada para um determinado projeto não é uma tarefa simples, pois pode existir uma falta de entendimento acerca das técnicas disponíveis, podendo acarretar na seleção de

uma técnica imprópria para o projeto e consequentemente podendo levar à falha do mesmo (KAUSAR *et al.,* 2010). Além disto, é importante ressaltar que a escolha da técnica varia de acordo com o problema a ser tratado, a equipe que trabalhará no desenvolvimento do sistema e o domínio do sistema (KOTONYA E SOMMERVILLE, 1998).

Quanto ao domínio dos sistemas críticos de segurança, é possível afirmar que são necessários requisitos que tragam precisão e confiabilidade. Portanto, é pertinente a utilização de técnicas de elicitação práticas e avançadas que sejam capazes de obter requisitos da melhor maneira (KAUSAR *et al.,* 2010), de preferência que sejam combinadas com técnicas de identificação de perigos.

Ignácio e Vavassori (2020) realizaram uma revisão sistemática da literatura com o objetivo de identificar todas as técnicas de elicitação de requisitos. Foram identificadas 96 técnicas de elicitação de requisitos em 30 artigos. No entanto, 59% dessas técnicas possuem apenas 1 citação, enquanto 21% das técnicas são encontradas em mais de 5 publicações.

A Figura 12 apresenta uma nuvem de palavras com todas as técnicas identificadas no estudo de Ignáco e Vavassori (2020), representadas conforme frequência. Portanto, quanto maior a fonte do nome da técnica, mais vezes essa técnica foi citada na literatura.

Figura 12 - Nuvem de palavras das técnicas de elicitação conforme frequência

Fonte: Ignácio e Vavassori (2020)

A visualização da nuvem de palavras sugere que a técnica de elicitação mais utilizada são as entrevistas, seguidas pela protótipação, *Joint Application Development* (JAD), *brainstoming* e questionários. A seguir iremos detalhar estas cinco técnicas.

Nesta dissertação foi desenvolvida uma abordagem baseada em uma lista de perguntas que podem ser utilizadas tanto como roteiro para uma entrevista estruturada a ser realizada com *stakeholders* da área de segurança, como através da aplicação de um questionário para os engenheiros de requisitos e/ou de segurança.

2.3.1.1.1 Entrevistas

Esta é considerada a técnica de elicitação mais utilizada e essencial para o desenvolvimento de sistemas, pois é uma forma usual de comunicação entre pessoas (TORO, 2000). Além disto, Kausar *et al.* (2010) sugerem que esta é a técnica de elicitação mais eficaz, tendo em vista que pode gerar uma discussão aberta acerca de itens complexos.

Kotonya e Sommerville (1998) explanam que a entrevista permite que o engenheiro de requisitos lide com diferentes *stakeholders*, facilitando a obtenção dos requisitos necessários para o desenvolvimento do sistema. No entanto, esta não é uma técnica tão simples quanto aparenta, tendo em vista que os *stakeholders* podem não saber o que desejam, além de possuírem dificuldade para externalizar o conhecimento.

As entrevistas podem ser classificadas em três tipos:

Não estruturadas: Este tipo de entrevista não possui um roteiro pré-definido, portanto, as perguntas são realizadas de modo espontâneo, permitindo um fluxo natural de interação entre o entrevistado e o(s) entrevistado(s) (BATISTA, 2003)

Semiestruturadas: Este tipo de entrevista exige a preparação de um *checklist* possuindo um conjunto de perguntas ou de itens que devem ser realizados na entrevista. No entanto, a ordem e o mecanismo das perguntas não são previamente definidos, além disto, permite que o entrevistador possa explorar mais perguntas que ele considere importantes (BATISTA, 2003).

Estruturadas: Possui um conjunto de perguntas elaboradas antecipadamente, sendo assim, o entrevistador deve utilizar a mesma sequência de perguntas para todos os entrevistados (BATISTA, 2003).

Para realizar uma entrevista é importante que o engenheiro de requisitos tenha uma preparação, buscando formular perguntas introdutórias, conduzindo o entrevistado ao foco do assunto que será discutido (BATISTA, 2003). Além disto,

Raghavan *et al.* (1994) afirmam que para a realização das entrevistas é necessário o desenvolvimento de algumas habilidades sociais gerais, dentre elas, a capacidade de ouvir, permitindo que o entrevistador auxilie o usuário no entendimento e na exploração dos requisitos de software.

Yousuf e Asger (2015) listam as seguintes vantagens das entrevistas:

- É uma boa técnica de elicitação para utilizar com tópicos complexos;
- É rica em informações, possibilitando a obtenção de requisitos detalhados;
- Permite o esclarecimento das ambiguidades;
- Fornece uma visão geral do sistema;
- As perguntas geralmente são respondidas, então, a taxa de falta de resposta é baixa; e
- O entrevistador consegue realizar uma leitura da linguagem corporal do entrevistado, podendo analisar as emoções ou desconforto do mesmo.

Yousuf e Asger (2015) também listam as seguintes desvantagens desta técnica, sendo elas:

- Não envolve muitos stakeholders;
- Não é fácil de agendar um tempo para as entrevistas com todos os stakeholders;
- As restrições de custos não permitem a coleta de grandes amostras;
- A qualidade dos dados coletados está ligada à experiencia do entrevistador e de como ele vai lidar com a entrevista; e
- É uma técnica demorada e trabalhosa.

2.3.1.1.2 Prototipação

Sommerville (2011) define um protótipo como uma versão inicial de um sistema de software que é usado para demonstrar conceitos, experimentar opções de design e, geralmente, descobrir mais sobre o problema e suas possíveis soluções.

A realização da descoberta de requisitos através da prototipação oferece algumas vantagens, sendo elas:

• Permite que os *stakeholders* experimentem os efeitos de seus requisitos,

tendo em vista que os protótipos podem ser tocados e testados. Sendo assim, esta técnica permite que os *stakeholders* possuam uma compreensão mais fácil do sistema e de como funcionam os requisitos implementados (POHL, 2010);

• Pode estimular que os *stakeholders* desenvolvam novos requisitos para o sistema (POHL, 2010);

• Auxilia no estabelecimento das viabilidades e utilidades do sistema antes do desenvolvimento do sistema (KOTONYA E SOMMERVILLE, 1998);

• É uma maneira efetiva para o desenvolvimento de interfaces de usuários (KOTONYA, 1998); e

• É capaz de desenvolver testes de sistemas para posteriormente ser utilizado no processo de validação do sistema (KOTONYA E SOMMERVILLE, 1998).

A elicitação de requisitos através da utilização da técnica de prototipação oferece as algumas desvantagens, sendo elas:

• Pode causar uma prorrogação de cronograma, adiando a entrega do produto (KOTONYA E SOMMERVILLE, 1998);

• Pode ser oferecido incompleto, tendo em vista que algumas vezes eles podem simular somente a versão final do sistema (KOTONYA E SOMMERVILLE, 1998);

• Quando um *stakeholder* que examina o protótipo tem expectativas erradas, ele pode tirar conclusões falsas do uso de protótipos (POHL, 2010); e

• A prototipação de sistemas complexos pode consumir bastante tempo (YOUSUF E ASGER, 2015)

2.3.1.1.3 Joint Application Development (JAD)

De acordo com Toro (2000) a técnica denominada JAD (*Joint Application Development),* foi criada pela IBM em 1977 como alternativa para as entrevistas individuais. O JAD consiste em reuniões em grupo durante um período de 2 a 4 dias, que ajudam os clientes e usuários a formular problemas e discutir as possíveis soluções. De acordo com Raghavan *et al.* (1994) o JAD se baseia em quatro princípios, sendo eles: (1) Dinâmica de grupo, através das sessões em grupo; (2) uso de recursos audiovisuais com o objetivo de melhorar a comunicação e o entendimento; (3) utilização de um processo organizado e racional e (4) utilização de uma documentação padrão, preenchida e assinada por todos os participantes.

Para Toro (2000) o JAD apresenta algumas vantagens em relação às entrevistas individuais, sendo elas:

- Reduz o tempo gasto, tendo em vista que a opinião do cliente é ouvida em conjunto com a opinião dos especialistas;
- A documentação gerada é analisada por todos os *stakeholders* e não apenas os engenheiros de requisitos; e
- Envolve mais os clientes e usuários no desenvolvimento.

Yousuf e Asger (2015) também enumeraram algumas desvantagens desta técnica:

- Precisa ser planejado adequadamente, pois, caso contrário, poderá levar ao desperdício de tempo e recursos;
- Requer facilitadores treinados;
- Requer muito planejamento e esforço; e
- É uma técnica cara.

2.3.1.1.4 Brainstorming

De acordo com Raghavan *et al.* (1994), *brainstorming* é uma técnica de grupo simples para gerar ideias, permitindo que as pessoas participantes sugiram e explorem ideias em um ambiente livre de críticas ou julgamentos, esta técnica é indicada para grupos de 4-10 pessoas. Toro (2000) acredita que a utilização do *brainstorming* como técnica de elicitação de requisitos é capaz de gerar uma ampla variedade de visões do problema, já que é possível formulá-lo de diversas maneiras. Comparando com o JAD, o *brainstorming* tem a vantagem de ser de fácil aprendizagem e requerer pouca organização, além disto, pode ser realizado através de videoconferências (RAGHAVAN *et al.,* 1994).

Yousuf e Asger (2015) enumeraram algumas vantagens do *brainstorming*:

- Não necessita de muitos recursos, reduzindo o custo;
- Os participantes não precisam ser altamente qualificados;
- É fácil de implementar;
- Ajuda na geração de novas ideias;
- Ajuda na resolução de conflitos;
- Cada participante pode expor sua opinião e compartilhar ideias.

Assim como as vantagens, Yousuf e Asger (2015) também enumeraram algumas desvantagens do *brainstorming*, sendo elas:

- Não é adequado para resolver problemas importantes;
- Pode ser demorado, caso não seja organizado adequadamente;
- A quantidade de ideias que surgem no brainstorming não significa que elas possuem boa qualidade; e
- Caso os participantes não estejam prestando atenção, é possível que haja uma repetição de ideias.

2.3.1.1.5 Questionários

Questionários consistem na aplicação de várias perguntas escritas para uma determinada amostra de usuários. De acordo com Batista (2003) pode ser realizado com questões fechadas, através de questões de múltipla escolha ou questões abertas, ou seja, questões discursivas, permitindo que o usuário responda com suas próprias palavras. Os questionários devem ser utilizados quando existe um conhecimento prévio acerca do problema que será resolvido e um número grande de clientes, de modo que os questionários permitem a percepção de como funcionam certos aspectos do software (KOTONYA E SOMMERVILLE, 1998).

Yousuf e Asger (2015) listaram algumas vantagens da técnica de aplicação de questionários para o levantamento de requisitos:

- Os questionários permitem alcançar um maior número de stakeholders em um curto período;
- É econômico;
- É difícil de estar enviesado.

Yousuf e Asger (2015) também citam algumas desvantagens da aplicação da técnica de questionário, como:

- O engenheiro de requisitos não consegue obter mais esclarecimentos sobre o problema;
- As perguntas podem ser mal interpretadas;
- É possível que exista ambiguidade nas perguntas;
- Existe a possibilidade de que comentários úteis não sejam repassados

dos stakeholders para o engenheiro de requisitos;

- Para obter mais informações, é necessário que seja utilizado em conjunto com outras técnicas de elicitação, tais como, entrevistas;
- É utilizado apenas para softwares de uso geral.

2.3.2. Documentação de requisitos

De acordo com Kotonya e Sommerville (1998) os requisitos de sistema e software são documentados por meio de um documento formal com o objetivo de comunicar os requisitos aos *stakeholders*. Este documento pode ser estruturado das mais diversas maneiras, de acordo com o domínio do sistema que será desenvolvido. É comum a utilização de Linguagem Natural (LN) para especificação dos requisitos, no entanto, os requisitos em LN nem sempre são fáceis de entender. Wilkinson e Mavin (2015) afirmam que a escrita de requisitos em LN pode ocasionar em alguns problemas, tais como:

1. Ambiguidade (uma palavra ou frase tem dois ou mais significados diferentes);
2. Imprecisão (falta de precisão, estrutura e/ou detalhe);
3. Complexidade (requisitos compostos contendo subcláusulas complexas e/ou várias declarações interrelacionadas);
4. Omissão (requisitos ausentes, particularmente requisitos para lidar com comportamentos indesejados);
5. Duplicação (repetição de requisitos);
6. *Wordiness* / Excesso de palavras (uso de um número desnecessário de palavras);
7. Implementação inadequada (declarações de 'como', em vez de 'o quê'); e
8. Falta de testabilidade (requisitos que não podem ser verificados).

A utilização de técnicas de modelagem de requisitos, que permitem a formulação, estruturação e modelagem dos requisitos, permitindo uma melhor a compreensão e representação do conhecimento necessário para as fases iniciais da ER. Dentre as técnicas de modelagem existentes na literatura, pode-se citar o framework i* ou iStar (YU, 1995), baseado em objetivos, que permite uma melhor representação dos requisitos iniciais. Este framework foi desenvolvido para modelar os ambientes organizacionais e seus sistemas de informação e funciona através da

descrição dos sistemas em dois modelos: o de Dependência Estratégica (SD) e o Raciocínio Estratégico (SR). Enquanto o modelo de dependência estratégica (SD) descreve as relações de dependência entre os atores presentes no contexto organizacional, o modelo de Raciocínio Estratégico é utilizado para descrever os interesses dos *stakeholders* e como eles podem ser tratados nas diferentes configurações do sistema (YU, 1995). Nesta dissertação utilizaremos uma extensão da linguagem iStar para modelar requisitos, denominada iStar4Safety (Ribeiro, 2019b).

2.4 iStar4Safety

Ribeiro (2019b) definiu uma extensão do iStar voltada para a modelagem de requisitos iniciais de segurança, denominada iStar4Safety. Além da representação dos requisitos do sistema através da utilização dos construtores padrões do iStar, a iStar4Safety permite a representação gráfica dos requisitos iniciais de segurança, através da adição de outros construtores gráficos, específicos para o domínio, sendo eles: Objetivo de Segurança (*Safety Goal*), Tarefa de Segurança (*Safety Task*), Recurso de Segurança (*Safety Resource*), Perigo (*Hazard*) e o link Obstrui (*Obstructs*).

Para o desenvolvimento do iStar4Safety foram utilizados alguns conceitos relacionados à segurança, adaptados do trabalho de Vilela et al. (2017b) e Ribeiro (2019b), sendo eles: acidente, perigo, causa de perigo, condição ambiental, requisitos funcionais de segurança, estratégias de segurança, recursos e nível de impacto do acidente, conforme apresentados no Quadro 5.

Quadro 5 - Conceitos mapeados em iStar4Safety

Conceito	Representação no iStar4Safety	Observações
Acidente	-	Acidente não possui uma representação gráfica no modelo, uma vez que é uma consequência do perigo obstruir um Objetivo de Segurança.
Perigo	Especialização de Objetivo	O perigo não é intencional, portanto, é considerado um anti-objetivo, ou seja, não deseja que este objetivo seja concretizado.
Causa de Perigo	Especialização de Objetivo	É representado como um perigo-filho
Condição Ambiental	Especialização de Objetivo	Por ser uma causa do perigo, é representado como um perigo-filho
Requisito de Segurança Funcional / Estratégia de Segurança	Especialização de Tarefa e Recurso	Estratégia de segurança é utilizada para lidar com o perigo juntamente com os Requisitos Funcionais de Segurança. Ambos buscam retratar a estratégia de segurança que será utilizada para mitigar e/ou minimizar o perigo.
Relacionamento entre construtores relacionados à perigos	Link “obstrui” e os refinamentos E/OU	O link obstrui é utilizado para associar objetivos de segurança e perigos. Os perigos e as estratégias de segurança são associados através dos links de refinamento E/OU

Fonte: Ribeiro (2019b) e Vilela (2017b)

Para modelar os elementos relacionados à segurança Ribeiro (2019a) desenvolveu 5 construtores gráficos, sendo eles: *SafetyGoal* (Figura 14**Erro! Fonte e referência não encontrada.**), *Hazard* (Figura 13), *SafetyTask (Figura 16***Erro! Fonte de referência não encontrada.**), *SafetyResource* (Figura 15) e *Link Obstructs* (*Figura 17*).

Figura 13 - Elemento SafetyGoal

Fonte: Ribeiro (2019a)

Figura 14 - Elemento Hazard

Fonte: Ribeiro (2019a)

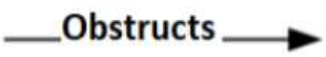
Figura 16 - Elemento SafetyTask

Fonte: Ribeiro (2019a)

Figura 15 - Elemento SafetyResource

Fonte: Ribeiro (2019a)

Figura 17 - Link Obstructs

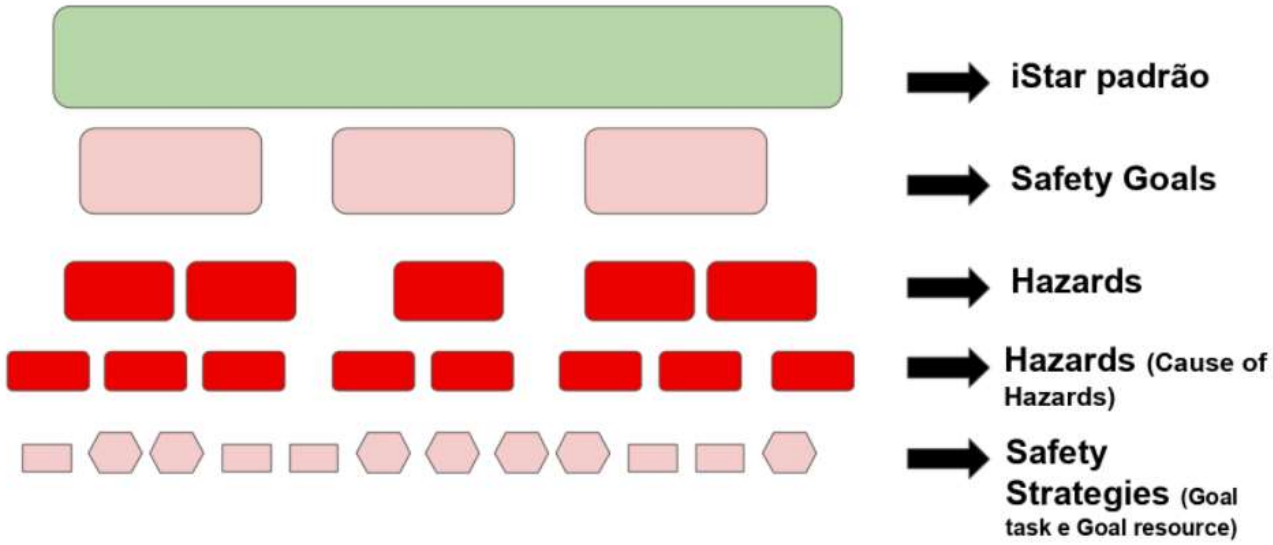


Fonte: Ribeiro (2019a)

De acordo com Ribeiro (2019a) um SafetyGoal é um objetivo crítico, que pode contribuir para a ocorrência de acidentes, caso algum perigo associado à ele ocorra. Já um *Hazard* é um obstáculo para que o *SafetyGoal* se concretize, ou seja, caso o *Hazard* aconteça, um acidente pode ocorrer. O link *Obstructs* liga um *Hazard* à um *SafetyGoal*. As *SafetyTasks* representam as ações seguras que um ator quer que sejam executadas a fim de mitigar um perigo. Por fim, o *SafetyResource* auxilia às *SafetyTasks*.

A modelagem em iStar4Safety segue a estrutura de uma árvore, em que a primeira camada é representada pelo iStar padrão, a segunda é representada pelos *SafetyGoals*, a terceira pelos *Hazards*, a quarta pelas *Cause of hazards* (utilizam o construto *Hazard)* e por fim as *SafetyStrategies*, representadas pelas *SafetyTasks* e *SafetyResources*, conforme representado pela Figura 18.

Figura 18 - Visão em Camadas - iStar4Safety



Fonte: Ribeiro (2019b)

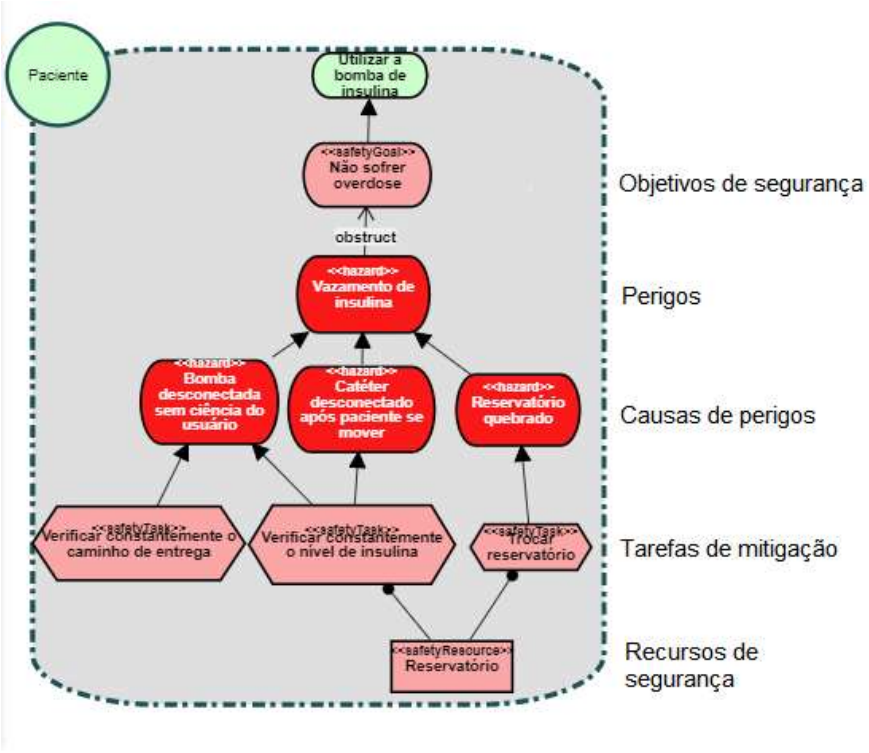
Um exemplo da modelagem em camadas pelo iStar4Safety é apresentado na Figura 19, que ilustra o refinamento de um objetivo do ator “Paciente” em um sistema de Bomba de Infusão de Insulina. O ator tem como objetivo principal utilizar a bomba de insulina, seu objetivo de segurança é não sofrer overdose, o perigo que impede que este objetivo seja satisfeito é o vazamento de insulina e pode ter três causas: (1) bomba desconectada sem ciência do usuário, (2) cateter desconectado após o paciente se mover e (3) reservatório quebrado. Para mitigar a causa Bomba desconectada sem ciência do usuário são realizadas as seguintes tarefas de segurança: verificar constantemente o caminho de entrega e verificar constantemente o nível de insulina. Quanto à causa Cateter desconectado após o paciente se mover, pode ser mitigada através da tarefa de segurança: verificar constantemente o nível de insulina, que tem como recurso de segurança o reservatório. Por fim, a causa Reservatório quebrado pode ser mitigada através da tarefa de segurança: trocar reservatório, com apoio do recurso de segurança Reservatório.

A abordagem Elicit4Safety busca permitir a modelagem em iStar4Safety a partir das diretrizes formuladas por Ribeiro (2019b) sendo elas:

- Modelar as funcionalidades relacionadas ao iStar 2.0 Padrão (parte do tipo não-segurança);
- Modelar o objetivo de segurança;

- Inserir todos os perigos para o objetivo de segurança modelado;
- Identificar todas as causas para cada perigo identificado;
- Definir as estratégias de mitigação para cada perigo folha;e
- Associar a estratégia de mitigação a um ator que ficará responsável por sua realização.

Figura 19 - Demonstração da modelagem em iStar4Safety



Fonte: Adaptado de Ribeiro (2019b)

2.5 TRABALHOS RELACIONADOS

Alguns trabalhos relacionados nos auxiliaram na aquisição do conhecimento necessário para o desenvolvimento do nosso projeto de pesquisa.

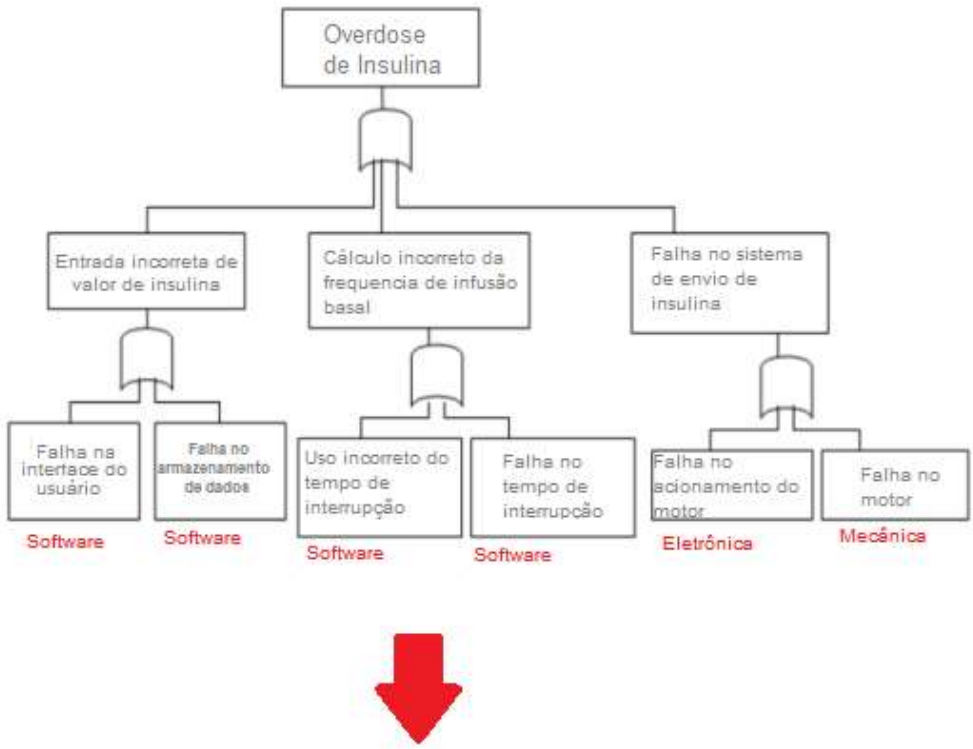
A literatura apresenta poucas técnicas de elicitação de requisitos no domínio de Sistemas Críticos de Segurança. Pode-se citar três técnicas: Du *et al.* (2014), Martins e De Oliveira (2014) e Provenzano *et al.* (2017), bem como o framework proposto por Yeow e Kia Chiam (2014) que apresenta as técnicas de análise de segurança e em qual etapa da elicitação de requisitos elas podem ser utilizadas.

A proposta de Du *et al.* (2014) é baseada na utilização de cenários e permite o refinamento da análise de segurança em comportamentos de software através de cenários específicos. Ela utiliza um método de análise de segurança, a *Fault Tree*

Analysis (FTA) utilizado para descobrir requisitos de segurança combinado com cenários, de modo que se torne possível que a análise de segurança expresse com precisão os estados perigosos do software. A técnica de elicitação de requisitos desenvolvida por Du *et al.* (2014) é bem complexa e utiliza lógica temporal juntamente com o FTA, se tornando uma limitação para os engenheiros de requisitos sem experiência em segurança.

O trabalho de Martins e De Oliveira (2014) tem como objetivo derivar requisitos funcionais de segurança a partir da construção de uma Árvore de Análise de Falhas (FTA). A técnica desenvolvida pelos autores apresenta alguns passos: primeiramente, é necessário identificar todas as situações de perigo do sistema em análise. Em seguida, deve ser construída uma árvore de falhas para a situação de perigo escolhida. Cada folha da árvore de falhas deve ser classificada de acordo com a causa da falha, podendo ser: software, eletrônica ou mecânica. Por fim, a cada falha identificada devem ser descritos requisitos funcionais de segurança do tipo "*Should*" ou *"Should not*", buscando mitigar as falhas encontradas e, consequentemente, o perigo. A Figura 20 apresenta um exemplo de utilização da técnica, onde os requisitos funcionais de segurança são derivados a partir da árvore de falhas (FTA).

Figura 20 - Derivação de Requisitos Funcionais de Segurança a partir do FTA



		Requisitos Funcionais de Segurança	
Requisitos de segurança	**Causas de falha**	**Requisitos "Should"**	**Requisitos "Should Not"**
Entrada incorreta de valor de insulina	Falha na interface do usuário (Software)	1.O sistema deve delimitar a faixa de valores de insulina durante o especificação do perfil de infusão basal. 2. A entrada do valor da insulina na especificação do perfil de infusão basal deve ser feita usando os botões para cima e para baixo.	1. O sistema não deve permitir que o usuário escolha um valor de insulina fora da faixa especificada de segurança para o perfil de infusão basal
	Falha no armazenamento de dados (Software)	3. O sistema deve armazenar pelo menos 3 perfis de infusão basal, cada um especificando 24 valores de insulina (um por hora)	
Cálculo incorreto da frequencia de infusão basal	Uso incorreto do tempo de interrupção (Software)	4. O sistema deve usar a função de interrupção de tempo oferecida pelo microcontrolador para contar o tempo	2.O sistema não deve continuar a infusão basal se o cronômetro é executado mais de 5 vezes por hora
	Falha no tempo de interrupção (Software)	5. O sistema deve verificar se a interrupção do tempo está funcionando conforme especificado pelo usuário.	3. O sistema não deve continuar a infusão basal se o erro acumulado entre o tempo real e o tempo especificado for superior a 0,1%.
Falha no sistema de envio de insulina	Falha no acionamento do motor (Eletrônica)	6. O sistema deve implementar procedimento de reconhecimento para confirmar se o motor está funcionando corretamente.	4. O acionador do motor não deve falhar durante a infusão de bolus.
	Falha no Motor	7. O sistema deve adotar um motor de alta precisão para administrar a insulina.	5. O motor não deve falhar durante a a infusão de bolus.

Fonte: Adaptado de Martins e de Oliveira (2014)

Ademais, a técnica desenvolvida por Martins e considerada uma técnica de fácil compreensão, que complementa a análise FTA, uma vez que, de acordo com Martins e de Oliveira (2014), apenas a utilização do FTA não traz informações suficientes para derivar os requisitos. No entanto, para aplicar esta técnica é

necessário que o engenheiro de requisitos possua um bom conhecimento no domínio de aplicação, uma vez que será necessário identificar as possíveis falhas para construção do FTA e o modo de mitigação delas.

Já Provenzano *et al.* (2017) fornecem uma abordagem heurística para o levantamento de requisitos de segurança baseada em uma ontologia. Inicialmente, devem ser levantados os perigos de um determinado sistema, para que possam ser classificados de acordo com uma ontologia denominada *Hazard Ontology* (HO). A partir da identificação e classificação destes perigos, é iniciada a abordagem heurística para a elicitação dos requisitos de segurança. Para que os engenheiros de requisitos utilizem essa técnica, é necessário que possuam um conhecimento prévio na *Hazard Ontology*, para que possam classificar corretamente os perigos descobertos e elicitar os requisitos de segurança.

Yeow e Kia Chiam (2014) propuseram um framework capaz de integrar técnicas de análise preliminar de segurança na atividade da elicitação de requisitos, atrelando essas técnicas com as etapas da elicitação de requisitos. A fim de dar suporte nesse framework, os autores desenvolveram uma aplicação capaz de auxiliar na identificação de qual técnica é a mais apropriada para ser utilizada em cada uma das três etapas de elicitação de requisitos abordadas pelos autores: pré-elicitação, meio da elicitação e pós-elicitção. Uma limitação deste trabalho é que ele apenas informa qual técnica de análise preliminar de segurança pode ser utilizada e em qual etapa da elicitação de requisitos, sendo assim, é necessário que o engenheiro de requisitos possua um conhecimento prévio acerca das técnicas para que elas sejam aplicadas corretamente.

O Quadro 6 apresenta um comparativo entre os trabalhos relacionados previamente demonstrados e a abordagem de elicitação de requisitos proposta por nós.

Quadro 6 - Comparação entre os trabalhos relacionados

	Elicit4Safety	Du et al (2014)	Provenzano et al (2017)	Martins e de Oliveira (2014)	Yeow e Kia Chiam (2014)
Baseada em perguntas	x				
Integração com iStar4Safety	x				
É apoiada por uma ferramenta	X				x
Baseada em ontologias			X		
Utiliza FTA		X		X	
Utiliza PHA	X				
Utiliza lógica temporal		X			
Baseada em cenários		X			

Fonte: Autora (2021)

O nosso trabalho diferencia-se desses, pois além de propor o desenvolvimento de uma abordagem de elicitação de requisitos de segurança, propõe também a integração desta abordagem com a linguagem ***iStar4Safety*** de modelagem de requisitos orientada à objetivos, uma vez que o iStar4Safety e suas *guidelines* para realizar a modelagem não são suficientes para a execução da atividade de elicitação de requisitos, já que se parte do pressuposto de que já foi realizada esta etapa e que os requisitos iniciais de segurança já foram previamente definidos.

Ademais, nós optamos por desenvolver uma nova abordagem e não reutilizar um dos trabalhos relacionados, pois não identificamos nos trabalhos previamente descritos definições que se adequassem ao iStar4Safety. Também vale ressaltar que buscamos desenvolver uma abordagem que apresentasse fácil usabilidade e

que fosse fácil de compreender, sem que seja necessário o conhecimento de lógica temporal ou das mais variadas técnicas de análise preliminar de segurança.

Pensando nisso, optamos por uma proposta que é constituída por uma lista de perguntas, que podem ser utilizadas tanto como roteiro para entrevistas estruturadas, quanto no preenchimento de um questionário disponível na ferramenta criada por nós para dar suporte a abordagem. Além disto, nossa abordagem possui perguntas que também auxiliam no entendimento do ambiente de desenvolvimento do SCSs, os atores envolvidos em sua utilização e os perigos atrelados a estes sistemas.

2.6 CONCLUSÃO DO CAPÍTULO

O capítulo 2 apresentou uma revisão dos fundamentos gerais utilizados no desenvolvimento deste trabalho. Primeiramente foi apresentado sobre segurança, seus conceitos e as técnicas de análise preliminar de segurança: FTA, HAZOP e PHA, uma vez que embora essas técnicas não foquem em engenharia de requisitos, apresentam um pontapé inicial para elicitação de requisitos, uma vez que através delas nós conseguimos visualizar os perigos do sistema e, a partir de então, elicitar os requisitos de segurança capazes de mitigar esses perigos. O próximo tema tratado foi o de Sistemas Críticos de Segurança e exemplos de falhas nestes sistemas. A Engenharia de Requisitos foi apresentada na seção 2.3, focando nas etapas de elicitação e documentação de requisitos. Também foram apresentadas as técnicas de elicitação de requisitos mais comuns. Na Seção 2.4 apresentamos a notação de modelagem de requisitos denominada iStar4Safety, que irá auxiliar a abordagem criada nesta dissertação. Por fim na Seção 2.5 foram relatados os principais trabalhos relacionados a esta dissertação.

3 ELICIT4SAFETY – ABORDAGEM PARA ELICITAÇÃO DE REQUISITOS INICIAIS DE SEGURANÇA EM SISTEMAS CRÍTICOS DE SEGURANÇA

Este capítulo apresenta a abordagem Elicit4Safety, que consiste em um conjunto de perguntas que visam auxiliar na descoberta dos objetivos de segurança, perigos, causas de perigos e consequentemente nos requisitos iniciais de segurança para posterior modelagem na linguagem iStar4Safety.

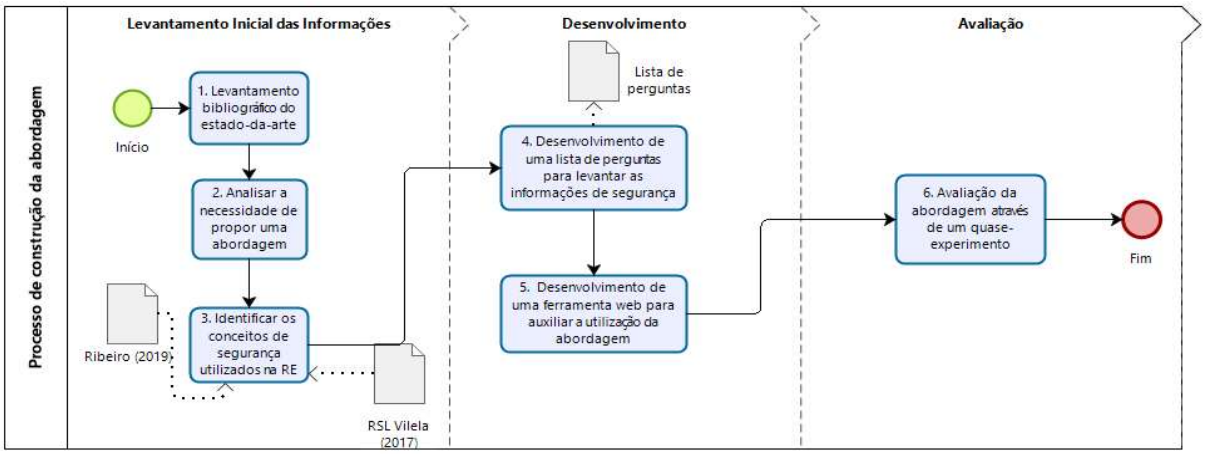
A seção 3.1 apresenta o processo para construção da abordagem. Nesta seção, são apresentadas as etapas realizadas para o desenvolvimento da abordagem Elicit4Safety. A seguir, na seção 3.2 é apresentado a abordagem Elicit4Safety, bem como a ferramenta desenvolvida para auxiliar na utilização da abordagem.

A seção 3.3 demonstra as diretrizes que devem ser utilizadas para utilizar a abordagem corretamente e para converter as informações obtidas no relatório da ferramenta para iStar4Safety. Por fim, a seção 3.4 apresenta um exemplo de utilização da abordagem.

3.1 PROCESSO DE CONSTRUÇÃO DA ABORDAGEM

O processo para construção da abordagem foi dividido em algumas fases: levantamento inicial das informações, desenvolvimento da abordagem e avaliação. A Figura 21 ilustra o processo de construção do método através da notação *Business Process Modeling and Notation* (BPMN), apresentando as fases e atividades realizadas.

Figura 21 - Processo de Construção – Elicit4Safety



Fonte: Autora (2021)

A Figura 21 demonstra as 3 fases para o desenvolvimento da abordagem e suas respectivas atividades. Na fase do levantamento inicial das informações, primeiro foi realizado um levantamento bibliográfico acerca do tema em estudo, posteriormente foi analisada a necessidade de propor uma abordagem e a última atividade consistiu na identificação de conceitos de segurança que podem ser utilizados na engenharia de requisitos, a partir do estudo da revisão sistemática de Vilela *et al.* (2017) e do trabalho de Ribeiro (2019b). A fase de desenvolvimento possui duas atividades: (1) o desenvolvimento de uma lista de perguntas utilizando os conceitos vistos anteriormente para levantar as informações de segurança do sistema e (2) desenvolvimento de uma ferramenta para auxiliar a utilização da abordagem. A última fase possui apenas uma atividade que se trata da avaliação da abordagem através de um quasi-experimento realizado com alunos de Universidade Federal de Pernambuco.

3.1.1 Levantamento inicial

O levantamento inicial das informações para desenvolvimento da abordagem proposta se deu através de uma revisão bibliográfica da literatura acerca do estado da arte das técnicas de elicitação de requisitos voltadas para o desenvolvimento de SCS. A partir da realização deste estudo, percebeu-se a escassez de técnicas de elicitação neste domínio. Portanto, buscando preencher esta lacuna, surgiu a necessidade do desenvolvimento de uma abordagem que possa integrar o levantamento e a modelagem de requisitos iniciais de segurança para esses sistemas. Foram selecionados dois trabalhos para auxiliar na obtenção de termos que seriam utilizados para a construção da abordagem, sendo eles: a Revisão Sistemática da Literatura (RSL) realizada por Vilela *et al.* (2017) que padroniza dos termos utilizados na ER e Engenharia de Segurança, por meio do desenvolvimento de quatro taxonomias. A taxonomia desenvolvida por Vilela *et al.* (2017) que trata das informações necessárias para descrição de um perigo permitiu a visualização de quais termos poderíamos utilizar para a realização de uma análise preliminar de segurança e consequentemente o levantamento dos requisitos capazes de mitigar e ou minimizar o risco inerente a esses perigos. O segundo trabalho que nos auxiliou na obtenção das informações acerca dos conceitos de segurança foi o de Ribeiro

(2019b) que apresenta as terminologias utilizadas para a construção da extensão iStar4Safety. Como o objetivo do Elicit4Safety é permitir a integração com a modelagem dos requisitos iniciais de segurança através da utilização do iStar4Safety, se faz necessário que os conceitos utilizados nos dois métodos estejam relacionados. Deste modo, a abordagem proposta nesta dissertação utilizou os conceitos relacionados à segurança mencionados por Ribeiro (2019b) e por Vilela *et al.* (2017) para o desenvolvimento de um conjunto perguntas que auxiliem no processo de realização de uma Análise Preliminar de Perigos (PHA) e que sejam suficientes para a posterior modelagem em iStar4Safety.

3.1.1.1 Levantamento bibliográfico do estado da arte

A primeira atividade da etapa de levantamento inicial das informações foi a realização de um levantamento bibliográfico do estado-da-arte das técnicas de elicitação de requisitos voltadas ao domínio de Sistemas Críticos de Segurança. O levantamento bibliográfico foi realizado nas seguintes bibliotecas digitais: ACM Digital Library, IEEE Xplore Digital Library, ScienceDirect e Springer. Foram procurados artigos e livros que tratassem acerca do tema em questão.

3.1.1.2 Analisar a necessidade de propor uma abordagem para aplicar a Engenharia de Requisitos em SCSs

A segunda atividade da etapa de levantamento inicial das informações buscou analisar a necessidade de propor uma abordagem para aplicar a ER em SCSs. O desenvolvimento desta atividade se deu após a identificação de uma lacuna na literatura acerca de técnicas de elicitação de requisitos voltadas ao domínio dos sistemas críticos de segurança. A partir de então, buscamos desenvolver uma abordagem capaz de suprir esta lacuna, visando também realizar a integração das etapas de elicitação e documentação de requisitos no contexto dos sistemas críticos de segurança.

De acordo com Gowen (1992) as principais causas de problemas de segurança de um sistema advêm da fase de especificação dos sistemas, uma vez que, muitos erros remetem aos documentos de especificação. Sendo assim, no desenvolvimento de SCSs é crucial que haja a prévia identificação dos perigos

atrelados a esses sistemas, para que os requisitos sejam especificados corretamente, visando mitigar e/ou diminuir o impacto desses perigos.

Ainda segundo Gowen (1992) a descoberta de perigos é semelhante à identificação de requisitos, inclusive, podemos utilizar técnicas de elicitação de requisitos adaptadas para auxiliar na identificação de perigos. Gowen (1992) demonstra quatro métodos para identificação de perigos, representados no Quadro 7.

Quadro 7 - Métodos para Identificação de Perigos

Método	**Descrição**
Análise de Dados Históricos	Identifica os perigos examinando os perigos e falhas de sistemas semelhantes.
Análise de protótipos	Usa um protótipo ou modelo para examinar cenários normais e anormais que podem causar perigos.
Brainstormings	Identifica perigos combinando um grupo de especialistas que podem contribuir com ideias espontâneas
Entrevistas	Uma conversa guiada com outra pessoa para obter informações sobre perigos

Fonte: Gowen (1992)

Uma entrevista é uma conversa guiada com uma outra pessoa, que tem como objetivo descobrir informações. A utilização das entrevistas é um método viável para identificação de perigos, pois permite que os engenheiros de requisitos iniciem uma entrevista com indivíduos com conhecimento em segurança para auxiliar na descoberta e perigos e/ou confirmar ou refutar possíveis perigos (GOWEN, 1992).

Pensando nisto, foi desenvolvida uma abordagem baseada em uma lista de perguntas que podem ser utilizadas tanto como roteiro para uma entrevista estruturada a ser realizada com *stakeholders* da área de segurança, como um questionário para os engenheiros de requisitos e/ou de segurança. Para suportar esta abordagem, desenvolvemos uma ferramenta que permite o preenchimento dos dados adquiridos na entrevista e/ou através da utilização de questionários. Espera-se que a abordagem aqui desenvolvida permita um correto levantamento dos perigos e requisitos iniciais de segurança visando a mitigação e a redução dos riscos

inerentes ao perigo, bem como a modelagem deles através de uma notação adequada.

3.1.1.3 Identificar os conceitos de segurança utilizados na ER

A terceira atividade da etapa do levantamento inicial das informações consistiu na identificação dos conceitos de segurança utilizados na Engenharia de Requisitos e foi realizada com base na leitura do artigo de Vilela *et al.* (2017) e na dissertação de Ribeiro (2019b) que tratam da engenharia de requisitos no âmbito da segurança dos sistemas. Foram identificados os conceitos representados no Quadro 8.

Quadro 8 - Conceitos de segurança utilizados na Engenharia de Requisitos

Conceito	Definição
Acidente (*Accident)*	Um evento indesejado e não planejado (porém não necessariamente inesperado) que resulta em, no mínimo, um nível específico de perda.
Perigo (*Hazard*)	Um estado ou conjunto de condições de um sistema, que juntos com outras condições ambientais do sistema, irão levar inevitavelmente a um acidente.
Causa de perigo (*Cause of hazard*)	É representada por uma condição que sozinha ou associada à outras, é/são suficiente(s) para o perigo relacionado à ela(s) ocorrer. As causas do perigo podem ser controladas ou até eliminadas em alguns casos.
Condições ambientais (*Environmental condition*)	É representada por uma condição que sozinha ou associada à outras, é/são suficiente(s) para o perigo relacionado à ela(s) ocorrer. As causas do perigo podem ser controladas ou até eliminadas em alguns casos.
Requisitos funcionais de segurança (*Functional safety requirements)*	São os requisitos funcionais usados para mitigar ou prevenir os efeitos de falhas identificadas na análise de segurança.
Nível de impacto do acidente (*Accident level impact*)	Este conceito define o quão crítico é um acidente em relação à segurança do sistema. É classificado através de cinco categorias: 1. Catastrófico (*Catastrophic)*; 2. Muito Severo (*Hazardous/Severe-major)*; 3. Considerável *(Major)*; 4. *Pequeno (Minor)*; 5. Sem efeito *(No effect)*.
Estratégias de segurança (*Safety strategies)*	São ações que visam mitigar as consequências de um possível acidente. O objetivo dessas

	ações é eliminar ou reduzir o risco associado a uma situação perigosa. Cada mitigação tem um custo para sua realização, que na maioria das vezes envolve o consumo algum recurso.
Recursos de segurança (*Safety resources*)	Na linguagem iStar recursos são apresentados como entidades informacionais ou físicas que são requeridas pelo ator à fim de realizar uma tarefa. Em Sistemas Críticos de Segurança, recursos são os ativos necessários para o correto funcionamento de requisitos críticos.
Risco (*Risk)*	Risco é o nível do perigo combinado com a probabilidade do perigo levar à um acidente (dano) e a exposição, ou duração, ao perigo (latência).
Dano (*Harm)*	Um dano pode ser causado como consequência do efeito de um perigo.
Tipo de dano (*Harm type)*	Um dano possui um tipo, representado por um ativo de um sistema, podendo ser: 1. Pessoas - Danos às pessoas podem ser: morte, perda, lesões, doenças, entre outros. 2. Propriedade - Danos à propriedade podem ser: destruição, roubo, acesso não autorizado ou divulgação não autorizada. 3. Ambiente – Danos ao ambiente podem ser: destruição, perda de uso. 4. Serviço – Danos ao serviço podem ser: corrupção, uso não autorizado (roubo), perda acidental de serviço ou negação de serviço.

Fonte: Adaptado de Ribeiro (2019b) e Vilela (2017)

3.1.2 Desenvolvimento

Nesta etapa foi realizada a construção da abordagem Elicit4Safety, através da realização de duas atividades: o desenvolvimento de uma lista de perguntas para levantar as informações de segurança e o desenvolvimento de uma ferramenta para auxiliar na utilização do método.

3.1.2.1 Desenvolvimento de uma lista de perguntas para levantar as informações de segurança.

A quarta atividade do processo de construção da abordagem Elicit4Safety consistiu no desenvolvimento de uma lista de perguntas, que podem ser utilizadas como roteiro de entrevista ou como questionário para engenheiros de requisitos e/ou

de segurança. As perguntas têm o objetivo de auxiliar na descoberta dos objetivos de segurança dos atores, perigos que impedem que estes objetivos sejam satisfeitos, as causas desses perigos e os requisitos funcionais de segurança do sistema, capazes de mitigar os perigos.

A princípio, a ideia era construir um roteiro de entrevista, tendo em vista que é uma técnica bem difundida e de fácil aplicação, inclusive, Ignário e Vavassori (2020) identificaram através de uma revisão sistemática da literatura que entrevista é a técnica de elicitação mais comum, sendo encontrada em 83% dos 30 artigos estudados pelos autores. Mas ao construir a ferramenta *web* para suportar a abordagem notou-se que ela pode ser utilizada tanto para dar suporte na realização de uma entrevista estruturada, quanto como um questionário preenchido pelo próprio engenheiro envolvido no desenvolvimento do SCSs.

Para o desenvolvimento de nosso roteiro de entrevista unimos os conceitos utilizados no iStar4Safety, bem como informações relevantes acerca da organização que irá utilizar os sistemas críticos. O conjunto de perguntas é dividido em cinco etapas, sendo elas: (1) Definindo o perfil do usuário e do cliente, (2) Definindo o perfil da empresa, (3) Descobrindo os detalhes do projeto, (4) Definindo os detalhes dos atores do sistema. O detalhamento das perguntas está disponível na seção 3.2.1.

3.1.2.2 Desenvolvimento de uma ferramenta para auxiliar a utilização da abordagem

A quinta atividade da construção da abordagem consistiu no desenvolvimento de uma ferramenta web para auxiliar na utilização da abordagem. Ela utiliza tecnologias JavaScript, CSS e HTML e pode ser acessado através do seguinte link: https://cin.ufpe.br/~sdm2/Elicit4Safety/. Maiores informações sobre seu uso são apresentadas na seção 3.2.2.

3.1.3 Avaliação

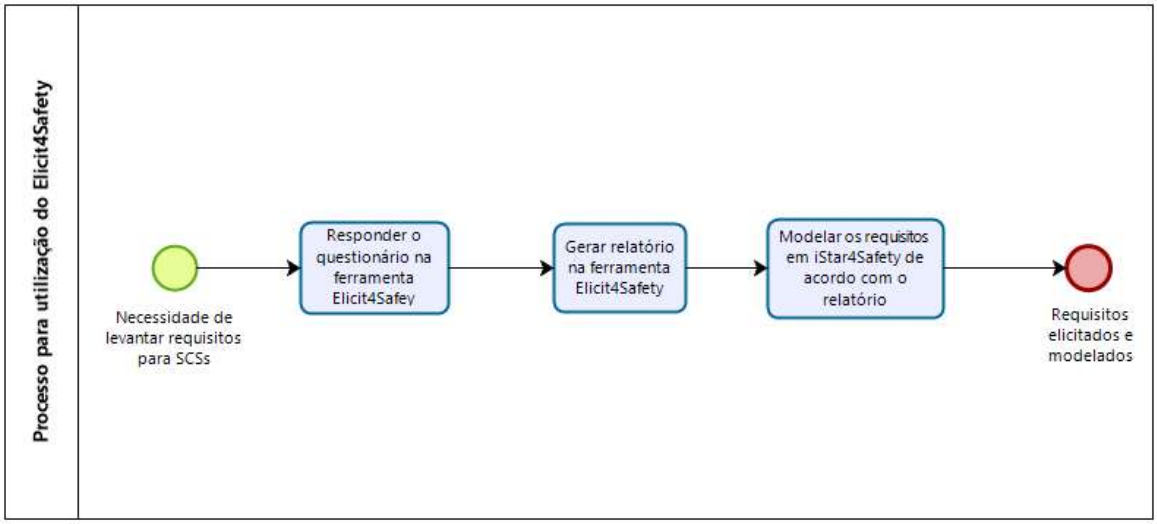
A etapa de avaliação consistiu na realização de uma atividade que foi a avaliação da abordagem através da realização de um quasi-experimento com alunos da graduação da disciplina de Especificação de Requisitos e Validação de Sistemas e de alunos da pós-graduação da disciplina Engenharia de Requisitos. No Capítulo 4

apresentamos a metodologia utilizada e as hipóteses testadas, bem como os resultados da avaliação.

3.2 A ABORDAGEM ELICIT4SAFETY

O Elicit4Safety consiste em um conjunto de perguntas associado à Análise Preliminar de Perigos (*Preliminary Hazard Analysis - PHA*) e a notação para modelagem de requisitos iniciais de segurança iStar4Safety. A fim de auxiliar na utilização da abordagem, desenvolvemos uma ferramenta de fácil utilização para que sejam preenchidas as informações inerentes ao sistema e a organização que está o desenvolvendo. A Figura 22 apresenta o processo para utilização da abordagem Elicit4Safety.

Figura 22 - Processo para utilização do Elicit4Safety



Fonte: Autora (2021)

Como pode ser visualizado na Figura 22 o processo para utilização do Elicit4Safety é bem simples, consistindo em apenas 3 atividades: responder as perguntas constantes através da ferramenta Elicit4Safety, gerar um relatório e a partir das respostas obtidas nele, realizar a modelagem dos objetivos de segurança, perigos, causas de perigos e estratégias de segurança através da notação iStar4Safety, seguindo as diretrizes constantes na seção 3.3.1.

3.2.1 Conjunto de perguntas utilizados na abordagem Elicit4Safety

O conjunto de perguntas desenvolvido, utiliza termos que se alinham com a

notação de modelagem de requisitos iStar4Safety e que permitem a realização de uma PHA. As perguntas são divididas em 4 categorias: (1) Definindo o perfil do *stakeholders*; (2) Definindo o perfil da empresa; (3) Descobrindo os detalhes do projeto; (4) Descobrindo os detalhes dos atores do sistema. As três primeiras categorias envolvem perguntas que foram utilizadas no questionário para avaliação do módulo de segurança do modelo de maturidade Uni-REPM (Vilela, 2018b). A última categoria de perguntas, para se adequar na modelagem iStar4Safety, possui um formato de árvore, possuindo algumas perguntas dentro de outras, por isso foi necessário o desenvolvimento de uma ferramenta para auxiliar e fazer com que as informações sejam representadas de maneira mais clara.

As perguntas e suas respectivas categorias são descritas abaixo:

1. **Definindo o perfil do stakeholder:** Neste conjunto de perguntas é identificado o perfil da parte interessada.
    1.1 Qual seu nome?
    1.2 Em qual organização você trabalha?
    1.3 Qual a sua função no desenvolvimento de sistemas críticos de segurança?
    1.4 Há quanto tempo você trabalha com sistemas críticos de segurança?
2. **Definindo o perfil da empresa:** Neste conjunto de perguntas é identificado o perfil da empresa em que o stakeholder trabalha.
    2.1 A empresa em que você trabalha pertence a qual domínio?
    2.2 A empresa em que você trabalha está estabelecida no mercado há quanto tempo?

3. **Definindo os detalhes do projeto:** Este conjunto de perguntas tem como objetivo descobrir os detalhes do projeto em que o stakeholder trabalha.
    3.1 Sobre o que é o seu projeto?
    3.2 Qual produto está sendo desenvolvido no seu projeto?
    3.3 Você possui alguma informação adicional que gostaria de compartilhar?

4. **Definindo os detalhes dos atores do sistema:** Este conjunto de perguntas tem como objetivo descobrir os detalhes dos atores do sistema, os perigos que o sistema apresenta, os acidentes que ele pode causar e os requisitos de segurança capazes de mitigar os perigos e prevenir os acidentes.
   4.1 Qual o nome do Stakeholder ou Sistema?
   4.2 Qual objetivo de segurança do Stakeholder / Sistema?
   4.3 Qual perigo impede que este objetivo de segurança seja concretizado?
   4.4 Qual seria o efeito (acidente) deste perigo?
   4.5 Qual nível de impacto deste efeito (acidente)?
   4.6 Se existir, quais as causas deste perigo?
   4.7 Quais tarefas de mitigação para esta causa?
   4.8 Quais recursos auxiliam a ação de mitigação?
   4.9 Quais tarefas de mitigação para este perigo?
   4.10 Quais recursos auxiliam a ação de mitigação?

Nós acreditamos que estas perguntas irão auxiliar o engenheiro de requisitos a encontrar e classificar os perigos oriundos de um sistema, colaborando com a realização de uma PHA e identificação dos requisitos iniciais de segurança dos SCSs. O conjunto de perguntas pode ser utilizado para conduzir uma entrevista com stakeholders do sistema ou preenchimento de um questionário através da ferramenta.

3.2.1.1 Checklist para avaliar a completude do modelo

A fim de assegurar a completude da abordagem, é necessário que o usuário verifique se preencheu todas as informações corretamente na ferramenta Elicit4Safety e se elas foram modeladas corretamente em iStar4Safety, para tanto, desenvolvemos um checklist, apresentando os seguintes itens:

- Todos os atores possuem objetivos de segurança?
- Todos os objetivos de segurança possuem perigos ligados a eles?
- Todos os perigos possuem causas e/ ou tarefas de mitigação?
- Todos os efeitos do perigo (acidente) foram preenchidos?

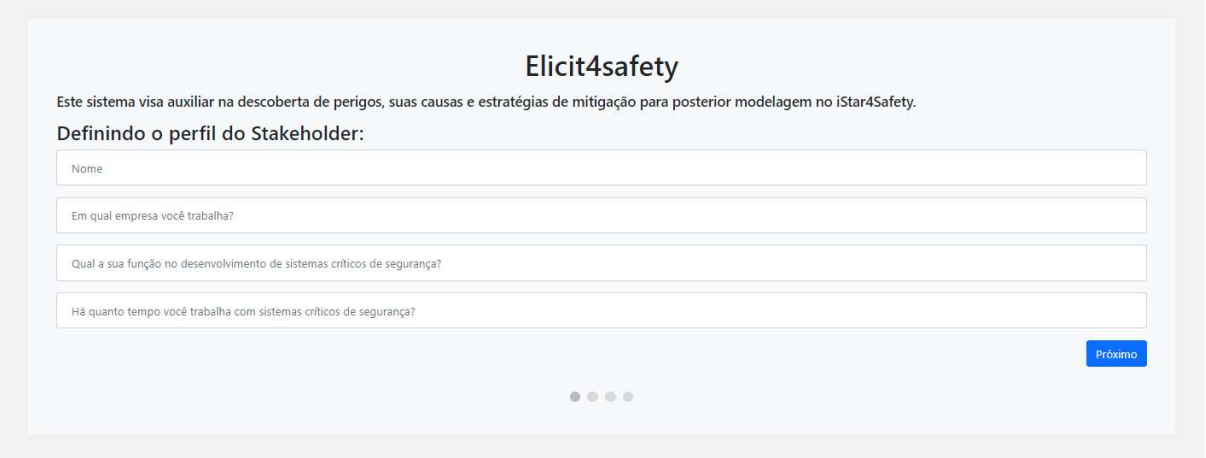
- Todos os efeitos do perigo (acidente) possuem o nível de impacto?

- Todos os campos do questionário foram preenchidos? Caso não, estão marcados como “não se aplica – N/A”?

3.2.2 Ferramenta para auxiliar na utilização da abordagem

A fim de auxiliar na utilização da abordagem Elicit4Safety, foi desenvolvida uma ferramenta web utilizando as tecnologias Java Script, CSS e HTML. A ferramenta é composta por 4 telas que, auxiliam no preenchimento das informações necessárias para a descoberta de requisitos funcionais de segurança e os elementos de segurança que serão mapeados em iStar4Safety bem como o elemento “efeito” que é utilizado durante a realização de uma Análise Preliminar de Perigos (PHA). As quatro telas da ferramenta referem-se às quatros etapas do conjunto de perguntas desenvolvido. Após o preenchimento dos dados, é possível gerar um relatório contendo todas as informações que foram preenchidas no sistema e salvá-lo em PDF. A Figura 23 representa a primeira tela da ferramenta e as perguntas que aparecem nela.

Figura 23 - Primeira tela da ferramenta Elicit4Safety



Fonte: Autora (2021)

A Figura 24 representa a segunda tela no sistema, constituída por perguntas referentes ao perfil da empresa que desenvolve o Sistema Crítico de Segurança.

Figura 24 - Segunda tela da ferramenta Elicit4Safety

Fonte: Autora (2021)

A Figura 25 apresenta a terceira tela no sistema, constituída por perguntas referentes aos detalhes do projeto do sistema crítico.

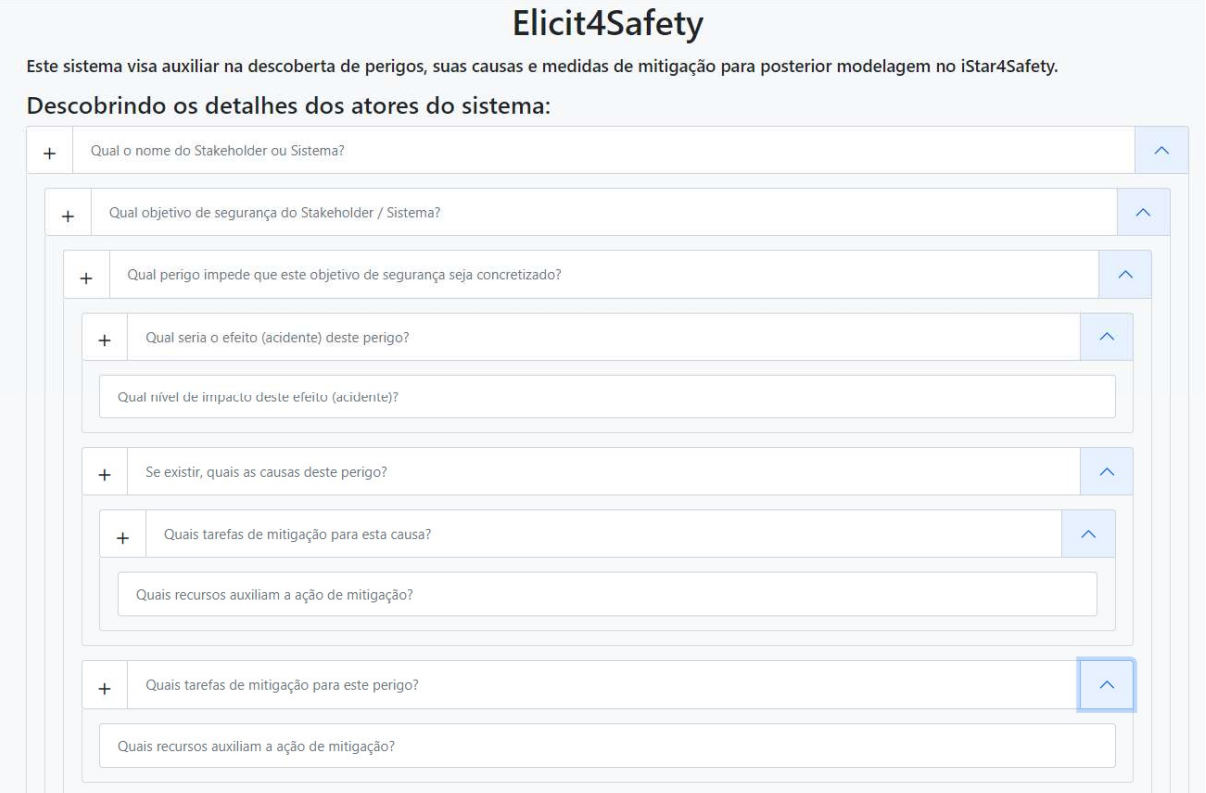
Figura 25 -Terceira tela da ferramenta Elicit4Safety

Fonte: Autora (2021)

A quarta tela, apresentada na Figura 26 é a mais detalhada do sistema e possui estrutura de árvore. Nela, estão as perguntas referentes aos atores do sistema e suas preocupações com segurança. Pode-se adicionar no sistema 1 ou N atores, e a partir de então é realizado o refinamento através da inclusão dos

objetivos de segurança, perigos que impedem que o objetivo de segurança seja realizado, o efeito do perigo e o nível de impacto, as causas do perigo -caso existam- e as estratégias de mitigação utilizadas, podendo ser representadas através de tarefas ou recursos de segurança. A ferramenta permite que possa ser adicionado mais de um elemento e que o mesmo possa ser refinado até o fim.

Figura 26 - Quarta tela da ferramenta Elicit4Safety (parte I)



Fonte: Autora (2021)

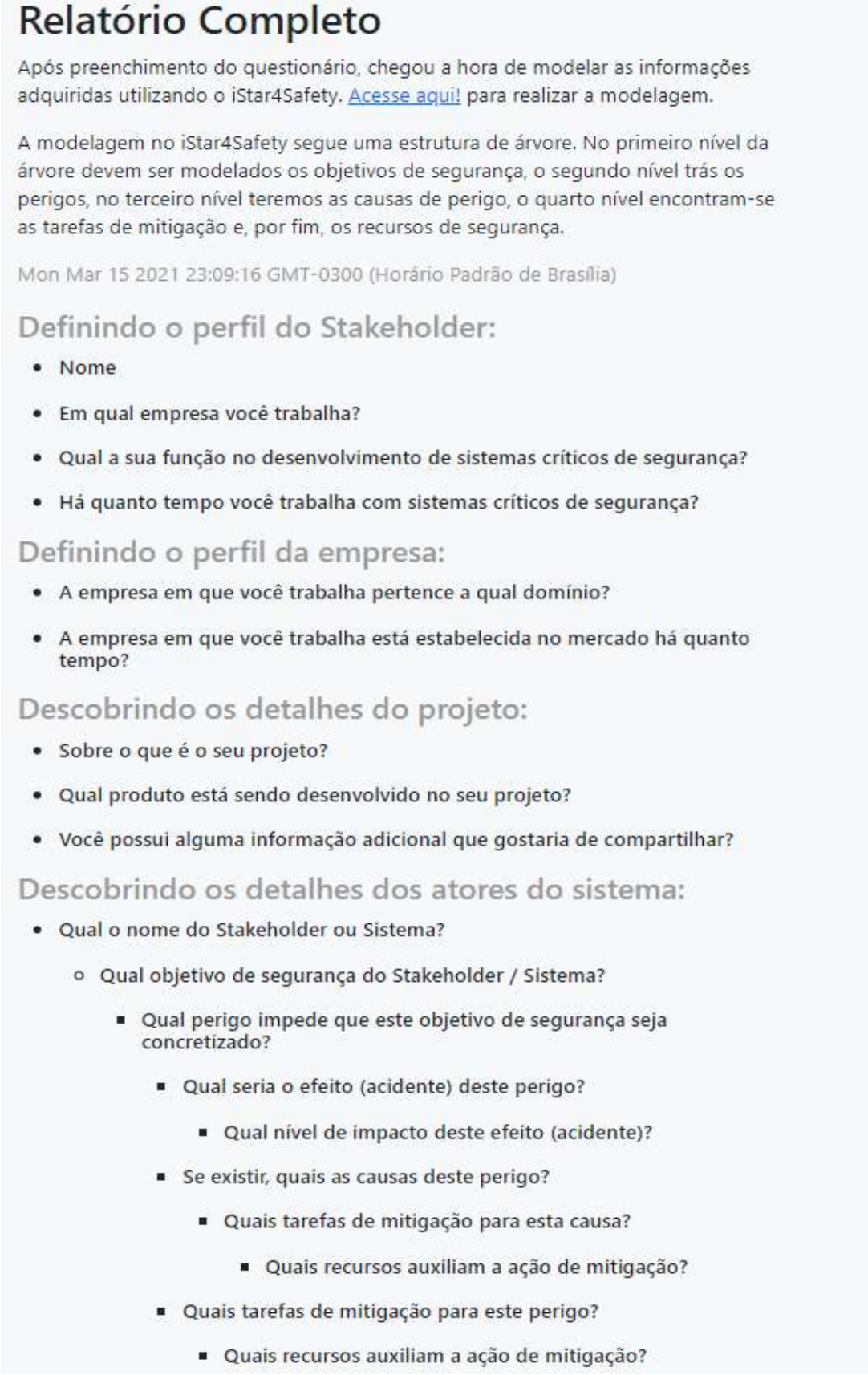
Na Figura 27 podemos observar como ficam as perguntas quando clicamos nelas para responder. O questionamento que na Figura 26 fica dentro do retângulo branco, ao clicarmos, vem para a parte de cima do retângulo acompanhada da representação gráfica do elemento referente a ela no iStar4Safety.

Figura 27 - Quinta tela da ferramenta Elicit4Safety (parte II)

Fonte: Autora (2021)

Por fim, a Figura 28 apresenta o relatório gerado pela ferramenta, nele é possível visualizar as informações preenchidas nas 4 telas e as respostas da quarta página representadas com tabulação que facilita a compreensão do nível de árvore. Ademais, o relatório possui um cabeçalho informando que após a finalização do preenchimento das informações, deve ser iniciado a modelagem em iStar4Safety e indexa o link da ferramenta pIstar4Safety, além de informar as dicas de como funciona a modelagem.

Figura 28 - Relatório gerado pela Elicit4Safety



Relatório Completo

Após preenchimento do questionário, chegou a hora de modelar as informações adquiridas utilizando o iStar4Safety. Acesse aqui! para realizar a modelagem.

A modelagem no iStar4Safety segue uma estrutura de árvore. No primeiro nível da árvore devem ser modelados os objetivos de segurança, o segundo nível trás os perigos, no terceiro nível teremos as causas de perigo, o quarto nível encontram-se as tarefas de mitigação e, por fim, os recursos de segurança.

Mon Mar 15 2021 23:09:16 GMT-0300 (Horário Padrão de Brasília)

Definindo o perfil do Stakeholder:

- Nome
- Em qual empresa você trabalha?
- Qual a sua função no desenvolvimento de sistemas críticos de segurança?
- Há quanto tempo você trabalha com sistemas críticos de segurança?

Definindo o perfil da empresa:

- A empresa em que você trabalha pertence a qual domínio?
- A empresa em que você trabalha está estabelecida no mercado há quanto tempo?

Descobrindo os detalhes do projeto:

- Sobre o que é o seu projeto?
- Qual produto está sendo desenvolvido no seu projeto?
- Você possui alguma informação adicional que gostaria de compartilhar?

Descobrindo os detalhes dos atores do sistema:

- Qual o nome do Stakeholder ou Sistema?
    - Qual objetivo de segurança do Stakeholder / Sistema?
        - Qual perigo impede que este objetivo de segurança seja concretizado?
            - Qual seria o efeito (acidente) deste perigo?
                - Qual nível de impacto deste efeito (acidente)?
            - Se existir, quais as causas deste perigo?
                - Quais tarefas de mitigação para esta causa?
                    - Quais recursos auxiliam a ação de mitigação?
            - Quais tarefas de mitigação para este perigo?
                - Quais recursos auxiliam a ação de mitigação?

Fonte: Autora (2021)

3.3 GUIA PARA UTILIZAÇÃO DA FERRAMENTA ELICIT4SAFETY

A fim de auxiliar no preenchimento das informações na ferramenta Elicit4Safety, desenvolvemos um guia para auxiliar na utilização da quarta etapa da ferramenta, uma vez que segue uma estrutura de árvore. Acreditamos que o passo a passo por nós elaborado possam permitir um melhor desempenho na hora da utilização da ferramenta. Os oito passos desenvolvidos são:

Passo 1: Insira o 1º ator do sistema na pergunta “Qual o nome do Stakeholder ou Sistema?”, caso haja mais de um ator deve ser incluído clicando no ícone “+” no lado esquerdo da pergunta, conforme demonstra a Figura 29.

Figura 29 - Exemplo para inserir mais de um ator na ferramenta Elicit4Safety

Fonte: Autora (2021)

Passo 2: Insira todos os objetivos de segurança para cada ator na pergunta “Qual objetivo de segurança do Stakeholder / Sistema?”. Caso haja mais de um objetivo de segurança deve ser adicionado clicado no ícone “+” do lado esquerdo da pergunta.

Passo 3: Insira todos os perigos para cada objetivo de segurança respondendo à pergunta “Qual perigo impede que este objetivo de segurança seja concretizado?”

Passo 4: Informe quais acidentes o perigo pode causar respondendo a pergunta “Qual seria o efeito (acidente) deste perigo?”

Passo 5: Em cada acidente, informe o nível de impacto ao responder à pergunta “Qual nível de impacto deste efeito (acidente)?”, devem ser considerados apenas os seguintes valores (adaptado de MIL-STD-882E):

1. Catastrófico (*Catastrophic)* (MAIOR IMPACTO): Pode resultar em um ou mais dos seguintes: morte, invalidez total permanente, impacto ambiental significativo irreversível ou perda monetária igual ou superior a 10 milhões;
2. Muito Severo (*Hazardous/Severe-Major*): Pode resultar em um ou mais dos seguintes: invalidez parcial permanente, lesões ou doença ocupacional que pode resultar na hospitalização de pelo menos três funcionários, impacto ambiental significativo ou perda monetária igual ou excedendo 1 milhão, mas menor que 10 milhões;
3. Considerável (*Major*): Pode resultar em um ou mais dos seguintes: lesão ou doença ocupacional resultando em um ou mais dias de trabalho perdidos, impacto ambiental moderado reversível ou perda monetária igual ou superior a $ 100k, mas inferior a 1 milhão;
4. Menor (*Minor*): Pode resultar em um ou mais do seguinte: lesão ou doença ocupacional não resultando em perda de um dia de trabalho, impacto ambiental mínimo ou perda monetária com menos de 100k;
5. Sem efeito (*No effect*) (SEM IMPACTO)

Passo 6: Reflita se o perigo possui causas. Caso sim, siga o passo 6.1 e caso não, siga o passo 6.2

Passo 6.1: Insira todas as causas do perigo na pergunta “Se existir, quais as causas deste perigo?” e refine-as inserindo as ações de mitigação como resposta na pergunta “Quais ações de mitigação para esta causa?” e recursos que auxiliam a ação de mitigação na pergunta “Quais recursos auxiliam a ação de mitigação?”

Passo 6.2: Caso o perigo não possua causas, insira as ações de mitigação que buscam mitigar o perigo na pergunta “Quais ações de mitigação para este perigo?” e refine inserindo os recursos de mitigação na pergunta “Quais recursos auxiliam a ação de mitigação?”

Passo 7: Verifique se você respondeu todas as perguntas corretamente. Caso tenha algum item que não se aplique você pode preencher o campo com “n/a” (não se aplica).

Ps.: Nem todas as tarefas de segurança possuem recursos auxiliando, então, caso seja necessário, preencha o campo de recurso com a informação “n/a” (não se aplica).

Passo 8: Finalize e gere o relatório. Você deverá utilizar as informações constantes no relatório para modelar em iStar4Safety.

Vale ressaltar que esse guia reflete apenas uma sugestão de etapas a serem seguidas. No entanto, dependendo da preferência do usuário da ferramenta, também pode-se utilizar uma abordagem puramente “*top-down*” refinando cada objetivo de segurança até chegar no seu nível mais baixo, representado pelos recursos de segurança e só então adicionar outro objetivo de segurança.

3.3.1 Guia para conversão das informações geradas no relatório em iStar4Safety

A nossa ferramenta ainda não é capaz de transformar automaticamente as informações recebidas em modelagem. Portanto, a realização da modelagem deve ser feita de maneira manual em iStar4Safety após preenchimento do questionário e geração do relatório. É importante ressaltar que primeiramente devem ser mapeados os elementos em iStar padrão, para que só depois sejam inclusos os elementos voltados à segurança.

Para auxiliar na atividade de conversão das informações, desenvolvemos um guia com sete passos, sendo eles:

Passo 1: Insira todos os atores inseridos como resposta na pergunta “Qual o nome do Stakeholder ou Sistema?”

Passo 2: Insira todos os objetivos de segurança inseridos como resposta na pergunta “Qual objetivo de segurança do Stakeholder / Sistema?” utilizando o elemento <<SafetyGoal>>.

Passo 3: Em cada objetivo de segurança modelado insira o valor da propriedade “accidentImpactLevel” inseridos como resposta na pergunta “Qual nível de impacto deste efeito (acidente)”?

Passo 4: Para cada objetivo de segurança modelado, insira os perigos que

impedem que este objetivo de segurança se concretize. Os perigos foram inseridos como resposta na pergunta “Qual perigo impede que este objetivo de segurança seja concretizado?” e devem ser modelados utilizando o elemento <<Hazard>>. Os perigos devem ser ligados aos objetivos de segurança com o link Obstruct.

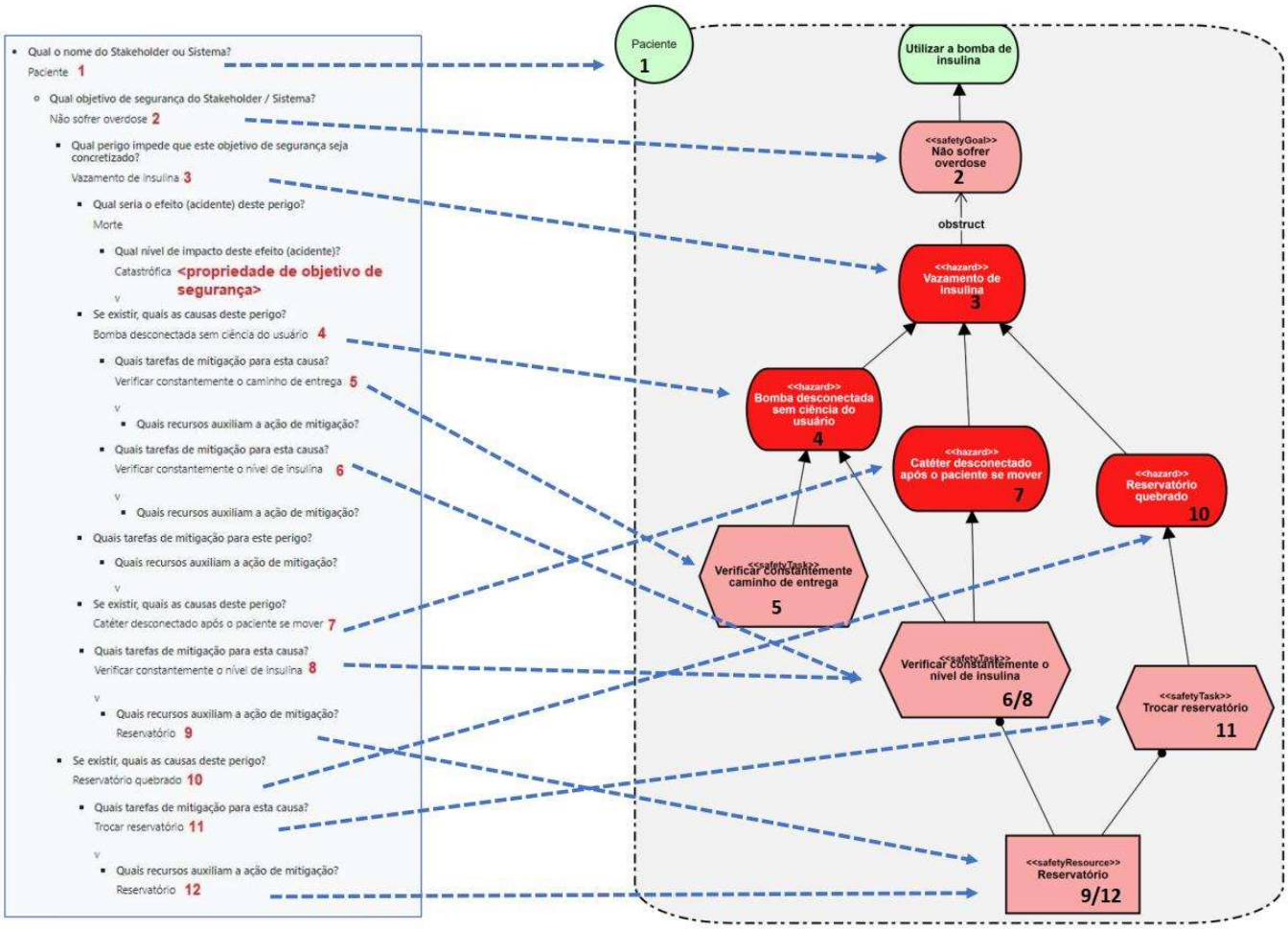
Passo 5: Para cada perigo informado, insira as suas causas. As causas também são representadas com o elemento <<Hazard>>, são consideradas “perigos-filho” e se ligam com o “perigo-pai” através dos links “AND” ou “OR”. As causas foram inseridas como resposta na pergunta “Se existir, quais as causas deste perigo”

Passo 6: Para cada causa, insira as tarefas de mitigação. As tarefas de mitigação (ou requisitos funcionais de segurança) são representadas através do elemento <<SafetyTask>>. As tarefas de segurança foram inseridas como resposta na pergunta “Quais tarefas de mitigação para esta causa?” e devem ser ligadas às causas através dos links “AND” ou “OR”.

Passo 7: Para cada ação de mitigação, insira os recursos que auxiliam a ação de mitigação. Os recursos foram inseridos como resposta da pergunta “Quais recursos auxiliam a ação de mitigação?”, são representados através do construtor <<SafetyResource>> e devem ser ligados às tarefas de segurança através do link “Needed-by”.

A Figura 30 apresenta um exemplo onde as informações geradas no relatório foram mapeadas em iStar4Safety e suas relações. O número 1 do relatório corresponde ao número 1 na representação gráfica e assim por diante. A Figura 30 refere-se a um Paciente de uma bomba de infusão de insulina.

Figura 30 - Relatório do Elicit4Safety mapeado em iStar4Safety

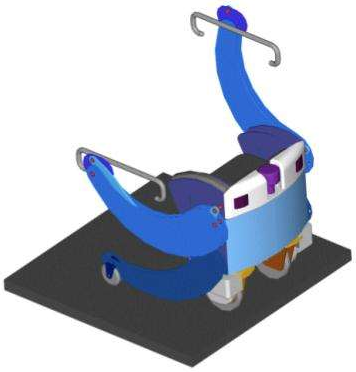


Fonte: Autora (2021)

3.4 EXEMPLO DE UTILIZAÇÃO DA ABORDAGEM

Essa subseção apresenta um exemplo de utilização da abordagem Elicit4Safety para descobrir e modelar os requisitos iniciais de segurança de um robô assistivo chamado MIRAS (MIRAS, 2009), representado na Figura 31. Ele tem como objetivo **auxiliar pessoas idosas a se levantarem, passearem e sentarem** quando não houver equipe médica disponível. O robô MIRAS consiste em uma **base de rodas e duas alças que simulam um guidão móvel e possui alguns sensores** com o objetivo de garantir o correto funcionamento do sistema. Para que o paciente seja levantado corretamente, é necessário que sejam inseridos os **dados do paciente**, indicando a **altura** que as alças do robô devem estar para que o paciente se apoie corretamente.

Figura 31 - Robô MIRAS



Fonte: Miras (2009)

A descrição completa do sistema robótico MIRAS a ser mapeado utilizando o Elicit4Safety está disponível no Anexo A.1.

Aqui iremos focar apenas na quarta etapa da abordagem, que diz respeito aos atores do sistema, perigos, causas de perigos e tarefas de segurança. Para realizar o exemplo, seguiremos o guia apresentado na seção 3.3.

3.4.1 Passo 1 - Inserir os atores do sistema

De acordo com o primeiro passo, devem ser inseridos os atores do sistema na pergunta “Qual o nome do Stakeholder ou Sistema?”. Neste exemplo, iremos focar apenas no ator Sistema Robótico, conforme mostrado na Figura 32. O exemplo completo está presente no Apêndice B.

Figura 32 – 1º Passo do Guia de Utilização da Abordagem

Fonte: Autora (2021)

3.4.2 Passo 2 - Insira todos os objetivos de segurança

O segundo passo do guia, solicita que sejam inseridos todos os objetivos de segurança para cada ator na pergunta “Qual objetivo de segurança do Stakeholder / Sistema?”. A Figura 33 apresenta um dos objetivos de segurança do stakeholder “Sistema Robótico” relacionado ao sistema robótico MIRAS, descrito no Apêndice A. No caso foi indicado que o objetivo de segurança seria *Não permitir que o paciente caia*.

Figura 33 - 2º Passo do Guia de Utilização da Abordagem

Fonte: Autora (2021)

3.4.3 Passo 3 - Insira todos os perigos para cada objetivo de segurança

O terceiro passo do guia de utilização da ferramenta, solicita que sejam inseridos todos os perigos para cada objetivo de segurança, através da resposta da pergunta “Qual perigo impede que este objetivo de segurança seja concretizado?”. Neste momento, iremos focar apenas em um perigo do ator sistema robótico, apresentado na Figura 34. No caso o perigo seria *Alças não estarem na altura correta do paciente.*

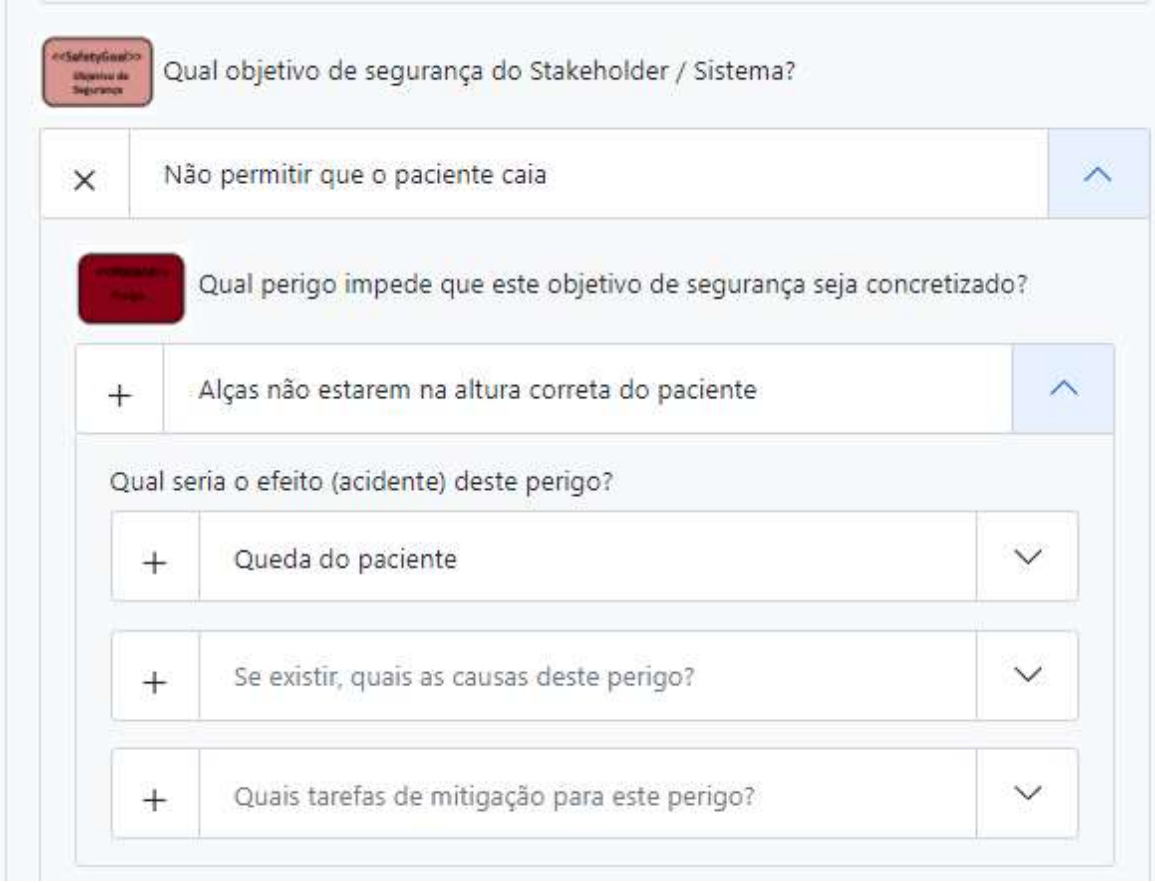
Figura 34 - 3º Passo do guia para utilização da abordagem

Fonte: Autora (2021)

3.4.4 Passo 4 - Informar quais acidentes o perigo pode causar

Nesta etapa devem ser informados os efeitos do perigo, ou seja, os acidentes que eles podem causar, respondendo à pergunta “Qual seria o efeito (acidente) deste perigo?” Neste exemplo, vamos refinar o perigo “Alças não estarem na altura correta do paciente”, do objetivo de segurança: Não permitir que o paciente caia. No caso o acidente seria *Queda do Paciente.* A Figura 35 demonstra o exemplo.

Figura 35 - 4º Passo do guia para utilização da abordagem



Fonte: Autora (2021)

3.4.5 Passo 5 - Informar o nível de impacto do acidente

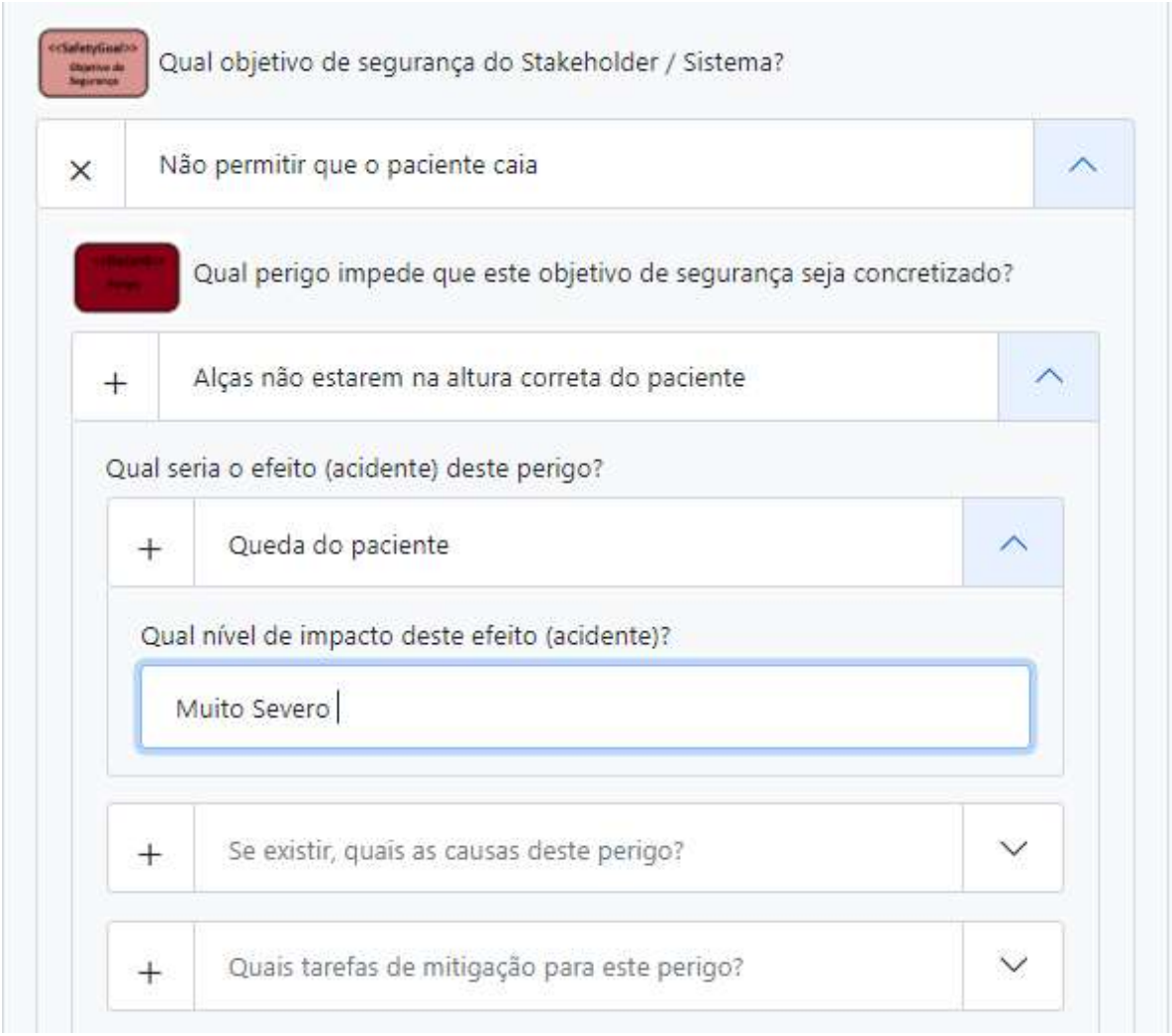
De acordo com o passo 5 do guia de utilização da abordagem Elicit4Safety, é necessário inserir o nível de impacto do acidente, ao responder à pergunta “Qual nível de impacto deste efeito (acidente)?”, de modo a considerar apenas os seguintes valores:

Catastrófico (*Catastrophic)* (MAIOR IMPACTO)
Muito Severo (*Hazardous/Severe-Major*)
Considerável (*Major*)
Menor (*Minor*)
Sem efeito (*No effect*) (SEM IMPACTO)

A Figura 36 apresenta o preenchimento da informação, e, como dito anteriormente, focaremos apenas no perigo “Alças não estarem na altura correta do

paciente”. Neste caso o impacto seria *Muito Severo.*

Figura 36 - 5º Passo do guia para utilização da abordagem



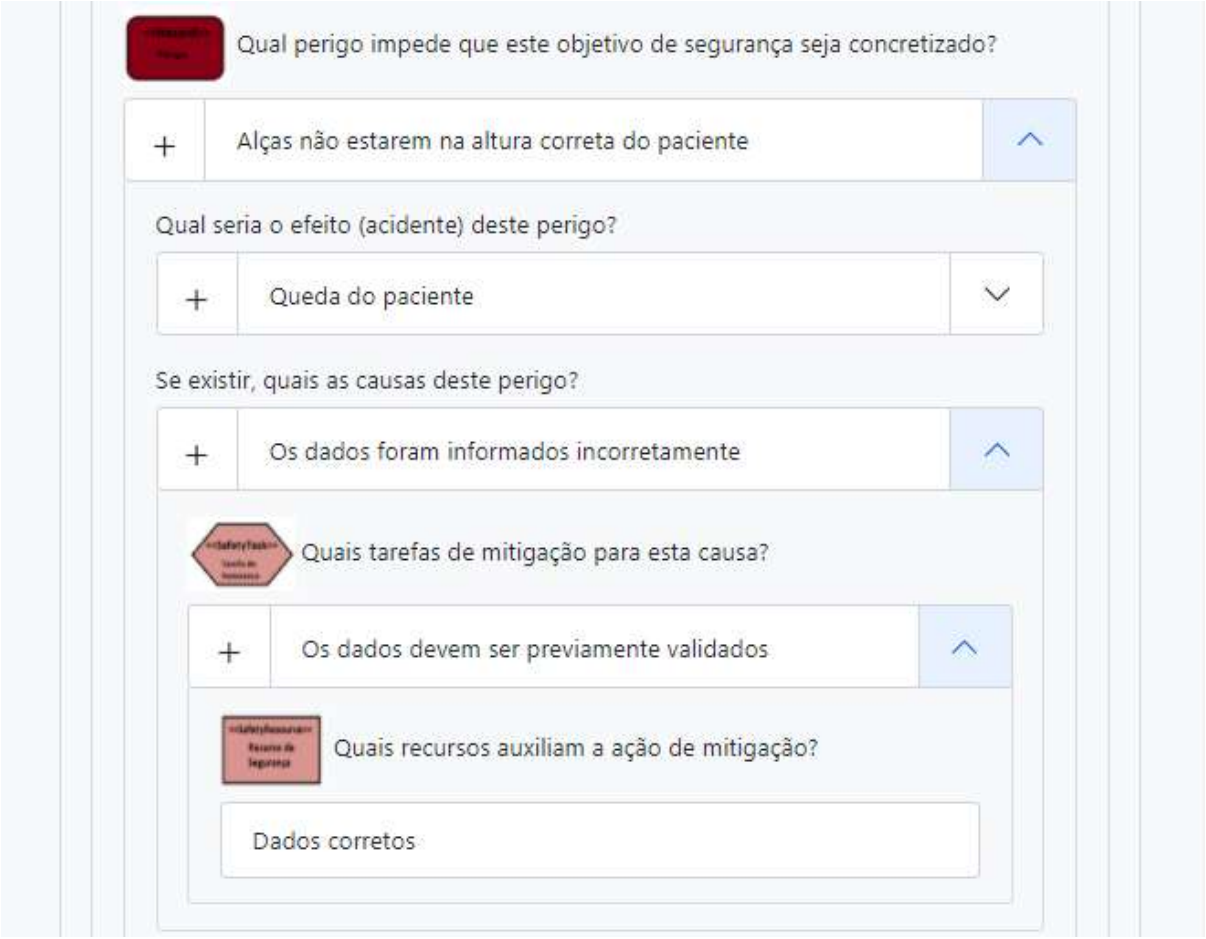
Fonte: Autora (2021)

3.4.6 Passo 6 – Refletir quanto às causas do perigo

O sexto passo do guia de utilização da abordagem é composto por 2 outros passos, que devem ser realizados de acordo com a existência ou não de causas de perigos. No exemplo aqui realizado, como o perigo possui causas, iremos seguir o passo 6.1, que diz o seguinte: Insira todas as causas do perigo na pergunta “Se existir, quais as causas deste perigo?” e refine-as inserindo as ações de mitigação como resposta na pergunta “Quais ações de mitigação para esta causa?” e recursos que auxiliam a ação de mitigação na pergunta “Quais recursos auxiliam a ação de mitigação?”.

A Figura 37 demonstra o exemplo de refinamento do perigo “Alças não estarem na altura correta do paciente”. Uma possível causa para este perigo poderia ser *Os dados foram informados incorretamente.* Para mitigar este perigo é proposta uma tarefa para validar os dados do paciente antes deles serem transferidos para o robô: *Os dados devem ser previamente validados.* Esta tarefa depende destas informações corretas estarem disponíveis, isto é, do recurso *Dados Corretos.*

Figura 37 - 6º Passo do guia para utilização da abordagem



Fonte: Autora (2021)

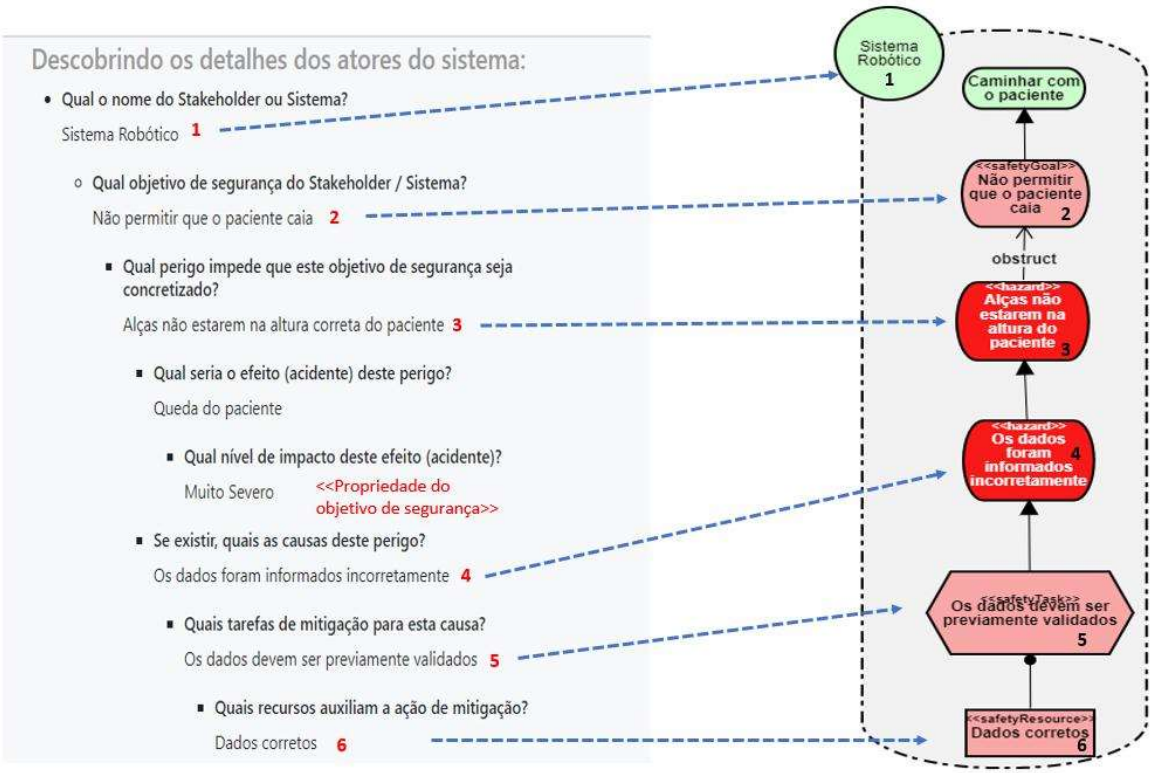
3.4.7 Passo 7 - Verifique se você respondeu todas as perguntas corretamente

Nesta etapa, deve ser conferido se todas as informações foram corretamente preenchidas nos campos correspondentes a elas. Após conferir tudo, o usuário pode gerar o relatório.

3.4.8 Passo 8 - Finalize e gere o relatório

Nesta última etapa o relatório com as informações inseridas no sistema deve ser gerado e a partir deste relatório deve-se iniciar a modelagem em iStar4Safety. A Figura 38 apresenta o relatório gerado a partir das informações adicionadas no sistema e a posterior modelagem delas em iStar4Safety.

Figura 38 - Relatório parcial do MIRAS Robot e a modelagem em iStar4Safety



Fonte: Autora (2021)

O mapeamento completo das informações de segurança do robô MIRAS encontra-se no APÊNDICE B.

3.5 CONCLUSÃO DO CAPÍTULO

Este capítulo apresentou a abordagem Elicit4Safety, desenvolvida com o intuito de auxiliar na descoberta de requisitos iniciais de segurança, estando alinhada com a técnica de modelagem iStar4Safety e a técnica de análise de perigos

PHA. A primeira seção do capítulo retratou o processo de construção da abordagem Elicit4Safety, a segunda seção focou em apresentar o Elicit4Safety, o conjunto de perguntas utilizado na abordagem e a ferramenta desenvolvida para dar suporte ao Elicit4Safety. A seção 3.3 apresentou um guia para utilização da quarta etapa da ferramenta e um guia para a conversão das informações geradas no relatório em iStar4Safety. A seção 3.4 apresentou um exemplo de utilização da abordagem.

O próximo capítulo focará em apresentar os resultados obtidos com a aplicação do quasi-experimento.

,

4 AVALIAÇÃO DO ELICIT4SAFETY

Este capítulo apresenta a avaliação da abordagem Elicit4Safety sendo dividido em seis seções. A primeira seção, retrata a definição do procedimento técnico experimental, ou seja, todo o planejamento do experimento realizado.

A seção 4.2 apresenta a análise e interpretação dos dados demográficos dos participantes do experimento, adquiridos após o preenchimento de um questionário demográfico (Disponível no Anexo A.3). A seção 4.3 dá ênfase a análise quantitativa da abordagem Elicit4Safety, através da avaliação de três variáveis dependentes: completude, número de elementos mapeados e tempo gasto para realizar a modelagem em iStar4Safety.

A seção 4.4 apresenta a análise qualitativa da abordagem Elicit4Safety. Para a realização desta etapa, o grupo experimental precisou responder um questionário (Disponível no Anexo A.4) após a utilização da ferramenta de apoio ao Elicit4Safety. Já a seção 4.5 apresenta as ameaças à validade do estudo. Por fim, a seção 4.6 apresenta a conclusão do capítulo.

4.1 DEFINIÇÃO DO PROCEDIMENTO TÉCNICO EXPERIMENTAL

A fim de avaliar a utilização do Elicit4Safety na descoberta dos requisitos iniciais de segurança e a modelagem na notação iStar4Safety, foi realizado um quasi-experimento com 18 alunos do Centro de Informática da Universidade Federal de Pernambuco dos níveis de graduação e pós-graduação que cursavam a disciplina de Especificação de Requisitos e Validação de Sistemas na graduação e a disciplina Engenharia de Requisitos na pós-graduação.

Para garantir o sucesso do experimento, o plano experimental foi realizado de acordo com o instrumento desenvolvido por Fonseca (2016), para realização de experimentos em Engenharia de Software que utilizam sujeitos experimentais humanos. O planejamento do quasi-experimento é apresentado a seguir.

4.1.1 Objetivos do estudo

O propósito deste estudo foi: **Analisar** a descoberta de requisitos de segurança em Sistemas Críticos de Segurança em conjunto com a abordagem Elicit4Safety **com o propósito de** avaliação, **com respeito à** completude, número

de elementos mapeados e tempo, **do ponto de vista de** engenheiros de software, **no contexto de** estudantes de graduação da disciplina de Especificação de Requisitos e Validação de Sistemas e estudantes da pós-graduação da disciplina de Engenharia de Requisitos do Centro de Informática (CIn) da Universidade Federal de Pernambuco (UFPE).

Para realização do quasi-experimento, os participantes foram divididos em dois grupos, um experimental e um de controle. No grupo experimental, foi solicitado aos participantes que utilizassem a abordagem Elicit4Safety para auxiliar na descoberta dos objetivos de segurança, perigos, causas de perigos e ações de mitigação de um sistema robótico e a posterior modelagem em iStar4Safety, enquanto os integrantes do grupo de controle deveriam modelar os aspectos de segurança do sistema robótico usando diretamente o iStar4Safety, isto é, sem auxílio da abordagem Elicit4Safety.

A fim de avaliar a opinião dos integrantes do grupo experimental sobre a abordagem e ferramenta Elicit4Safety, foi implementado um questionário com questões abertas e fechadas, que deveriam ser respondidas por eles, após a modelagem do sistema robótico, disponível no Apêndice A4.

4.1.2 Questões de pesquisa

Para avaliar a abordagem Elicit4Safety foram definidas as seguintes questões de pesquisa (QP), demonstradas no Quadro 9.

Quadro 9 - Questões de pesquisa e justificativa

Questão de pesquisa	**Justificativa**
• QP1: A utilização da abordagem Elicit4Safety tem efeito sobre a completude do mapeamento realizado em iStar4Safety?	Esta questão de pesquisa está diretamente relacionada com a variável dependente completude. Ela visa identificar se existem diferenças na completude da modelagem com e sem a utilização da abordagem Elicit4Safety.
• QP2: A utilização da abordagem Elicit4Safety tem efeito sobre o número de elementos mapeados em iStar4Safety?	Esta questão de pesquisa está diretamente relacionada com a variável dependente número de elementos mapeados, visando identificar se existem diferenças estatísticas significativas no número de elementos modelados utilizando e sem utilizar a abordagem Elicit4Safety.
• QP3: A utilização da abordagem Elicit4Safety tem efeito sobre o tempo para realizar a modelagem em iStar4Safety?	Esta questão de pesquisa está diretamente relacionada com a variável dependente tempo e visa identificar se existem diferenças estatísticas significantes no tempo levado para modelar com e sem a utilização do Elicit4Safety.

Fonte: Autora (2021)

4.1.3 Hipóteses

De acordo com Juristo e Moreno (2001) ao tomar uma decisão estatística, é necessário a construção de hipóteses sobre a população em questão. As hipóteses podem ser verdadeiras ou falsas, e durante a realização de um estudo experimental, as hipóteses são geradas de acordo com o objetivo do experimento. Sendo assim, para realização do quasi-experimento foram definidas as seguintes hipóteses:

- $H0_1$: A completude da modelagem de requisitos utilizando a abordagem

Elicit4Safety é igual à completude mapeando diretamente em iStar4Safety;

- $H0_2$: O número de elementos mapeados com a utilização da abordagem Elicit4Safety é igual aos gerados mapeando diretamente em iStar4Safety; e
- $H0_3$: O tempo utilizado para mapear com auxílio do Elicit4Safety é igual ao utilizado para mapear diretamente em iStar4Safety.

4.1.4 Tratamentos

Ao realizar um método experimental, como é o caso do quasi-experimento aqui realizado, são utilizadas comparações a fim analisar os fenômenos. Para tanto, é necessário definir uma variável independente que será avaliada e os tratamentos que irão incidir sobre ela, visando comparar os resultados obtidos com a utilização de cada tratamento. No caso desta pesquisa, foi escolhida a utilização da abordagem Elicit4Safety como o tratamento aplicado aos participantes do quasi-experimento, onde serão avaliados:

- Geração do modelo em iStar4Safety com utilização da abordagem Elicit4Safety; e
- Geração do modelo em iStar4Safety sem utilização da abordagem Elicit4Safety.

4.1.5 Variáveis

Esta subseção apresenta as variáveis definidas para execução do quasi-experimento, que de acordo com Easterbrook *et al.* (2008), deve-se possuir variáveis independentes, que deverão ser manipuladas para medir seu efeito e variáveis dependentes, que deverão ser testadas para visualizar como as variáveis independentes as afetam.

4.1.5.1 Variável independente

Conforme Juristo e Moreno (2001) as variáveis independentes delimitam o cenário onde o fenômeno é observado de maneira mais clara. Para tanto, neste estudo foi definida como variável independente a modelagem de requisitos iniciais de segurança.

4.1.5.2 Variáveis dependentes

De acordo com Juristo e Moreno (2001), o resultado de um experimento é denominado de variável dependente. Foram estabelecidas três variáveis dependentes, ou seja, aquelas que queremos medir, sendo elas:

- Completude na modelagem realizada no iStar4Safety;
- Número de elementos modelados no iStar4Safety;
- Tempo para modelagem no iStar4Safety.

4.1.6 Métricas

Foram definidas as seguintes métricas para dar suporte na quantificação das variáveis dependentes, sendo elas:

- M1 – Completude dos modelos gerados, medida através da utilização das diretrizes de completude formuladas por Ribeiro (2019b);
- M2 – Número total de elementos modelados em iStar4Safety (objetivos de segurança, perigos, tarefas de segurança e recursos);
- M3 – Tempo em minutos.

4.1.7 Participantes

Os participantes selecionados foram alunos de graduação da disciplina de Especificação de Requisitos e Validação de Sistemas e da disciplina de pós-graduação de Engenharia de Requisitos, ambas ministradas no Centro de informática da UFPE. Os participantes foram recrutados através de uma atividade no Google Sala de Aula e no dia do experimento participaram de uma reunião através do Google Meet, onde foram repassadas as informações do experimento. Para participação no experimento foi necessário concordar com o Termo de Consentimento Livre e Esclarecido (Disponível no Anexo A.2).

Participaram do quasi-experimento 18 alunos voluntários, ou seja, não foram oferecidos pagamentos e/ou outros tipos de vantagem. Para executar o quasi-

experimento, primeiro foi necessário que os alunos realizassem uma avaliação de conhecimentos acerca de Segurança (*Safety*) e iStar4Safety (Disponível no Anexo A.6), valendo 100 pontos, e apenas foram considerados na amostra os participantes que tiraram nota maior ou igual a 60. Também foi necessário que os participantes preenchessem um questionário demográfico (Disponível no anexo A.3), composto pelas seguintes informações: endereço de e-mail, idade do participante, sexo, curso, período, nível conhecimento em elicitação de requisitos, em Análise Preliminar de Perigos, modelagem iStar e em *Safety*. O grupo experimental recebeu um formulário extra a ser preenchido após realização da atividade, a fim de coletar a opinião dos alunos acerca da utilização da abordagem Elicit4Safety (Disponível no anexo A.4).

O estudo se iniciou com 24 alunos, no entanto 6 foram descartados da amostra pelos seguintes motivos:

- dois alunos não entregaram a atividade conforme solicitado;
- dois sequer entregaram a atividade;
- um entregou a atividade igual à de outro colega;
- um não atingiu a nota mínima requerida na avaliação de conhecimentos e não entregou a atividade conforme solicitado.

4.1.8 Materiais experimentais e tarefas a serem realizadas

Para realização do experimento, os participantes precisaram de um computador com acesso à internet para a realização das reuniões via Google Meet, preenchimento dos formulários no Google Forms e modelagem na ferramenta piStar4Safety[2].

Foram realizados 04 treinamentos que englobaram os assuntos requeridos para participação no quasi-experimento, sendo eles: iStar, Segurança (*Safety)* e seus conceitos, iStar4Safety e por fim o treinamento acerca da abordagem Elicit4Safety, que abrangeu apenas os participantes do grupo experimental.

[2] Disponível para acesso através do link: https://www.cin.ufpe.br/~jhcp/pistar/4safety/#

Os participantes do estudo realizaram uma atividade em que simulavam ser um engenheiro de requisitos e deveriam mapear os elementos de segurança em iStar4Safety do Sistema Robótico MIRAS (MIRAS, 2009). Para a realização da atividade os alunos receberam a descrição do sistema (conforme Anexo A.1), o modelo de Raciocínio Estratégico (SR) inicial em iStar a ser carregado na ferramenta piStar4Safety. Além disto, o grupo experimental recebeu um material de apoio com as diretrizes para preenchimento das informações no Elicit4Safety e como converter as informações obtidas no relatório na modelagem iStar4Safety (disponível na seção 3.3). Foi solicitado que os participantes enviassem o .txt do mapeamento realizado na ferramenta iStar4Safety, e os integrantes do grupo experimental também deveriam enviar o relatório em PDF gerado na ferramenta Elicit4Safety.

4.1.9 Design experimental

Será utilizado o desenho experimental "*Simple Randomised Designs: One Alternative per Experimental Unit*", que de acordo com Juristo e Moreno (2001) é o meio mais simples para comparar a variável dependente para cada unidade experimental, que possui apenas uma variável independente. Como possuímos apenas uma variável independente, escolhemos este desenho experimental, e cada participante ficou responsável por executar apenas um tratamento, através de uma divisão realizada de maneira aleatória. Embora os sujeitos experimentais tenham sido atribuídos de modo aleatório ao tratamento, a nossa amostra foi pré-definida, tendo em vista que foram chamados apenas estudantes da área de Computação, portanto, não é suficiente para se caracterizar como um experimento.

O Quadro 10 relata os participantes do quasi-experimento e o tratamento que cada um ficou responsável. Sendo assim, na Tabela 10, **P** representa os participantes, **EI** representa o grupo experimental Elicit4Safety e **i*S** representa o grupo de controle denominado iStar4Safety.

Quadro 10 - Participantes e o tratamento que ficaram responsáveis

P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18
EI	EI	i*S	i*S	i*S	i*S	i*S	EI	i*S	EI	EI	i*S	EI	i*S	EI	i*S	EI	i*S

Fonte: Autora (2021)

Inicialmente foram designados 12 participantes para cada grupo, no entanto, foram descartados da amostra 4 participantes do grupo experimental e 2 do grupo de controle, por não seguirem as diretrizes do quasi-experimento.

4.1.10 Ética

No desenvolvimento do presente estudo foram consideradas as questões éticas identificadas por Wohlin (2012), sendo elas:

- Consentimento informado: Este é um princípio ético essencial no desenvolvimento de pesquisas empíricas orientadas para humanos. De acordo com Wohlin (2012), os sujeitos deverão participar voluntariamente do estudo e deverão possuir informações suficientes para decidirem se irão ou não participar do quasi-experimento. Os sujeitos possuem a opção de se retirar do estudo a qualquer momento, sem qualquer penalidade. Para tanto, o pré-requisito para participação do quasi-experimento é o aceite do Termo de Consentimento Livre e Esclarecido.
- Confidencialidade: Para atingir o princípio ético da confidencialidade o presente estudo assegurou aos participantes a privacidade dos dados, anonimato dos dados e anonimato de participação.

4.2 ANÁLISE E INTERPRETAÇÃO DOS DADOS DEMOGRÁFICOS

Esta subseção discute os dados demográficos coletados através do questionário demográfico respondido por todos os participantes do quasi-experimento.

4.2.1 Perfil acadêmico dos participantes

Foram analisados os dados dos 18 sujeitos experimentais através do preenchimento de um questionário demográfico que visava possibilitar a análise tanto do perfil dos participantes, quanto o nível de conhecimento acerca dos temas abordados no quasi-experimento. O Quadro 11 mostra os dados do perfil acadêmico dos participantes.

Quadro 11 – Perfil Acadêmico dos Participantes

Participante	Curso	Nível	Período
#1	Ciência da Computação	Graduação	8º
#2	Ciência da Computação	Graduação	10º
#3	Ciência da Computação	Graduação	7º
#4	Ciência da Computação	Graduação	10º
#5	Ciência da Computação	Graduação	8º
#6	Engenharia da Computação	Graduação	9º
#7	Engenharia da Computação	Graduação	9º
#8	Ciência da Computação	Graduação	9º
#9	Ciência da Computação	Mestrado	2º
#10	Ciência da Computação	Doutorado	1º
#11	Ciência da Computação	Doutorado	1º
#12	Informática	Mestrado	2º
#13	Ciência da Computação	Graduação	6º
#14	Sistemas de Informação	Graduação	10º
#15	Engenharia da Computação	Graduação	9º
#16	Ciência da Computação	Mestrado	2º
#17	Ciência da Computação	Graduação	9º
#18	Engenria da Computação	Graduação	9º

Fonte: Autora (2021)

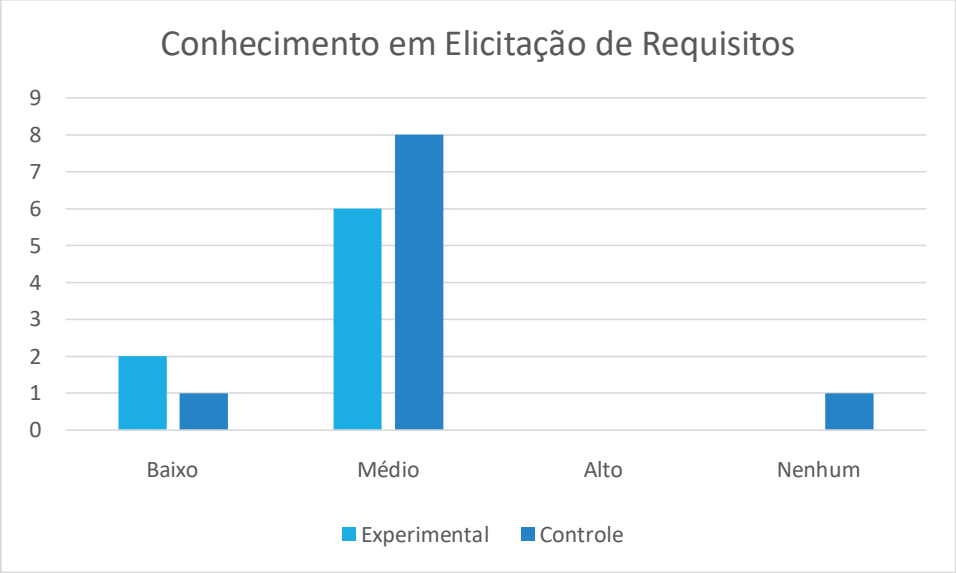
4.2.2 Nível de conhecimento

O nível de conhecimento dos participantes acerca de quatro assuntos relacionados ao experimento foi analisado: conhecimento em elicitação de

requisitos, análise preliminar de perigos, iStar e *Safety*.

A Figura 39 apresenta um gráfico de colunas com o nível de conhecimento dos participantes em Elicitação de Requisitos, tanto no grupo experimental, quanto no grupo de controle.

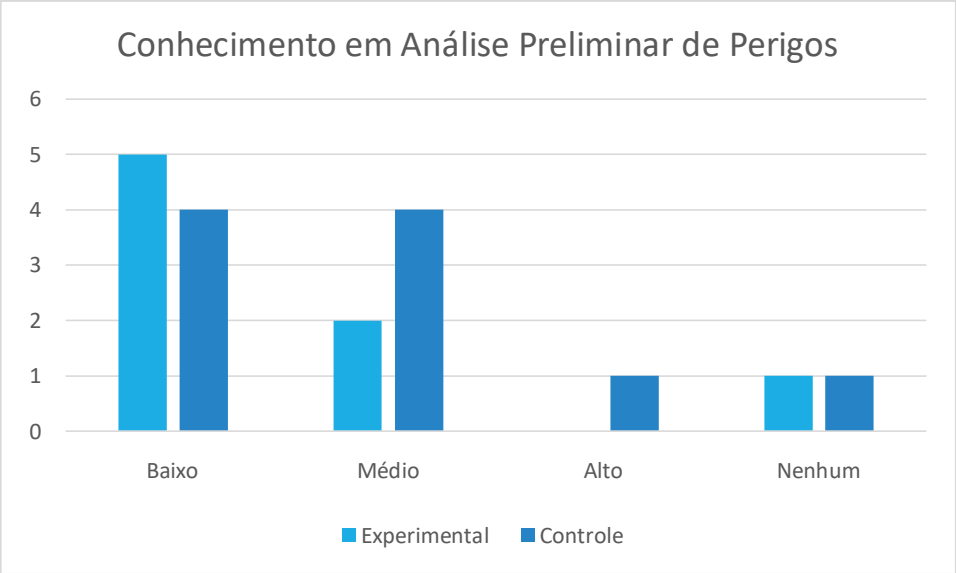
Figura 39 - Nível de conhecimento dos participantes em Elicitação de Requisitos



Fonte: Autora (2021)

A Figura 40 apresenta um gráfico de colunas com o nível de conhecimento dos participantes em Análise Preliminar de Perigos separados pelos grupos participantes do experimento.

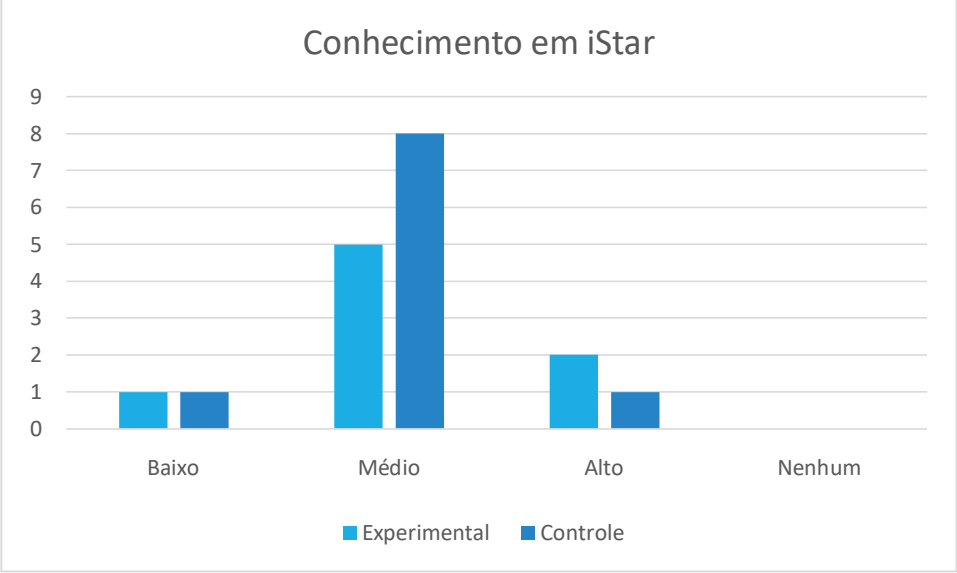
Figura 40 - Nível de conhecimento dos participantes em Análise Preliminar de Perigos



Fonte: Autora (2021)

A Figura 41 apresenta um gráfico de colunas com o nível de conhecimento dos participantes em iStar, de acordo com o grupo.

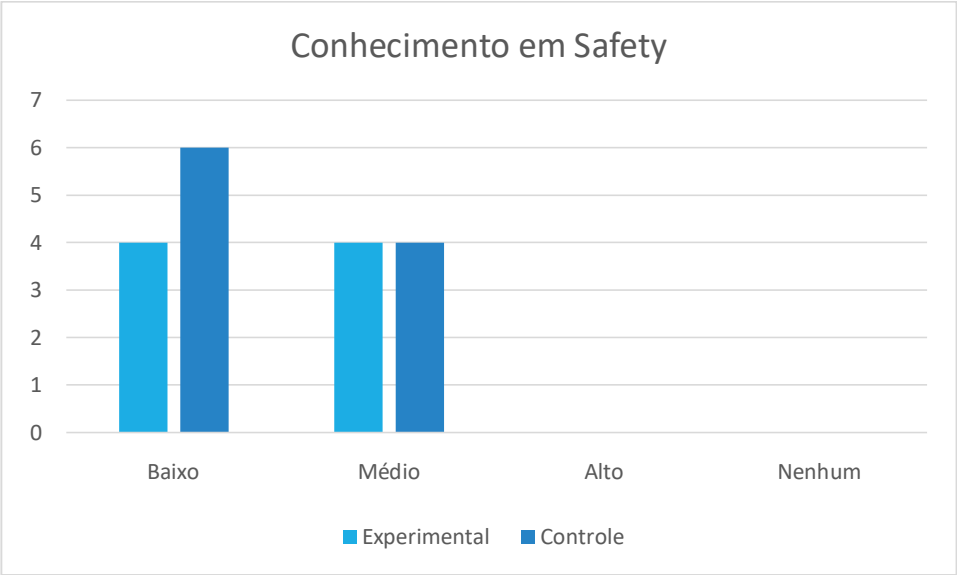
Figura 41 - Nível de conhecimento dos participantes em iStar



Fonte: Autora (2021)

A Figura 42 apresenta um gráfico de colunas com o nível de conhecimento dos participantes em *Safety,* de acordo com o grupo em que estavam alocados.

Figura 42 - Nível de conhecimento dos participantes em *Safety*



Fonte: Autora (2021)

4.3 ANÁLISE QUANTITATIVA DA ABORDAGEM ELICIT4SAFETY

Os dados quantitativos gerados a partir da modelagem em iStar4Safety realizado pelos participantes foram analisados através de estatística descritiva e testes estatísticos, com auxílio do software SPSS *Statistics Subscription* (IBM, 2020). Os dados brutos podem ser vistos no Apêndice C. Para realização das estatísticas descritivas utilizamos *boxplot* com o objetivo de analisar as diferenças entre os tratamentos em relação às medianas, primeiro quartil e terceiro quartil. A fim de interpretação dos boxplot, vale ressaltar que o grupo experimental foi denominado como Elicit4Safety e o grupo de controle foi denominado como iStar4Safety.

De acordo com Marôco (2018), a aplicação de um teste paramétrico depende das seguintes condições: que a variável dependente possua distribuição normal e que as variâncias sejam homogêneas. Sendo assim, para aplicação do teste t de Student, foi necessário primeiramente verificar se as amostras possuíam distribuição normal, através do teste de Shapiro-Wilk e se as variâncias eram homogêneas, através do teste de Levene. Como todos os valores de p obtidos com o teste Shapiro-Wilk (Quadro 11) deram maiores que 0.05, observa-se que existe uma distribuição normal entre os dados.

Quadro 11 - Resultados do teste Shapiro-Wilk

Variável Dependente	Valor de p
Completude	0,216
Elementos Mapeados	0,341
Tempo	0,231

Fonte: Autora (2021)

Na aplicação do teste de Levene (Quadro 12), os valores de p também deram acima de 0.05, indicando que os dados são homogêneos.

Quadro 12 - Resultados do teste de Levene

Variável Dependente	Valor de p
Completude	0,439
Elementos Mapeados	0,198
Tempo	0,898

Fonte: Autora (2021)

Deste modo, escolhemos a realização do teste t de Student de amostras independentes, baseada no fato de que os dados seguem uma distribuição normal, são homogêneos e temos poucas unidades experimentais. De acordo com Juristo e Moreno (2013) o teste t é um teste usual utilizado para analisar os dados obtidos de uma pequena amostra e para aplicar regras de decisão sobre a significância dos resultados.

A partir da realização do teste t, iremos obter o valor de T, graus de liberdade, diferença média, valor de P. O valor de t compara as médias entre os tratamentos e incorporam o tamanho amostral e a variabilidades nos dados, isso significa que caso o valor de t seja igual 0, indica que os resultados das amostras dos tratamentos aplicados são iguais. Para que seja possível interpretar o valor de t, é necessário se inserir um contexto maior, ou seja, as distribuições t, que por sua vez são definidas através de seus graus de liberdade, que é um valor relacionado ao tamanho da amostra. A diferença média indica o valor de diferença das médias entre os tratamentos, ou seja, o valor obtido pelo grupo de experimental menos o valor obtido pelo grupo de controle. Quanto ao valor de p calculado através da utilização do teste t, caso seja menor que 0.05 presumimos que existem diferenças significativas entre

os tratamentos.

4.3.1 Avaliação da completude

Para avaliar a porcentagem da completude dos modelos, foi proposta a questão de pesquisa QP1: “A utilização da abordagem Elicit4Safety tem efeito sobre a completude do mapeamento realizado em iStar4Safety?”. A métrica utilizada para calcular a porcentagem de completude foi baseada nas diretrizes de completude[3] propostas por Ribeiro (2019), sendo elas:

1. Todos os objetivos de segurança têm uma propriedade de nível de impacto do acidente com um valor inserido;
2. Todos os objetivos de segurança têm um ou mais perigos associados ou objetivos de segurança que nesse caso são seus refinamentos;
3. Todos os perigos-raiz estão ligados à um ou mais objetivos de segurança-folha pelo link *Obstructs*;
4. Todos os perigos-folha estão ligados à uma ou mais estratégias de segurança; e
5. Todas as estratégias de segurança estão associadas ao ator que as realizam, menos no caso de serem realizadas pelo próprio ator em que estão definidas.

Como possuem 5 diretrizes, atribuímos o valor de 20% para cada diretriz atingida, caso a diretriz seja satisfeita parcialmente, computamos 10% e caso não seja atingida, o valor não é computado.

O Quadro 13 mostra os resultados da análise estatística descritiva da completude dos modelos utilizando os dois tratamentos.

[3] Após a realização do experimento, nós desenvolvemos um checklist para avaliação de completude, presente na seção 3.2.2.1

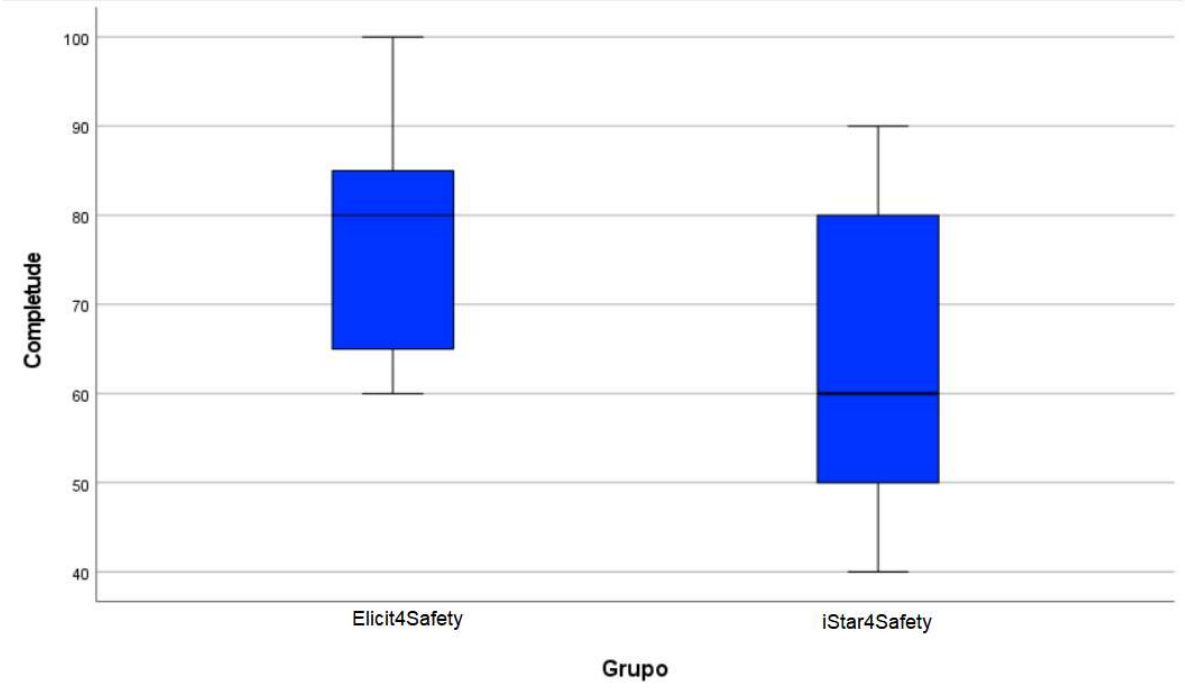
Quadro 13 - Análise Descritiva da Variável Completude

Completude		
	Média	Mediana
Elicit4Safety	77,5	80
iStar4Safety	65	60

Fonte: Autora (2021)

A Figura 43 mostra o *boxplot* da completude dos dois grupos: o grupo experimental, que utilizou a abordagem Elicit4Safety e o grupo de controle, que não utilizou a abordagem Elicit4Safety e mapeou as informações do sistema robótico MIRAS diretamente em iStar4Safety. A mediana, o primeiro quartil e o terceiro quartil mostram que a completude dos modelos elaborados pelo grupo experimental é maior do que a dos modelos elaborados pelo grupo de controle.

Figura 43 - Boxplot referente à Completude



Fonte: Autora (2021)

O Quadro 14 mostra os resultados da análise estatística através da realização do teste t. A diferença média entre os tratamentos (12,5) indica que o grupo experimental possuiu uma completude maior que o grupo de controle. No

entanto, o valor de p obtido (t(16) = 1,554; p=0,140) indica que não existem diferenças estatísticas significativas entre os tratamentos. Sendo assim, não foi possível rejeitar a $H0_{1:}$ *A completude da modelagem de requisitos utilizando a abordagem Elicit4Safety é igual à completude mapeando diretamente em iStar4Safety.*

Quadro 14 - Análise da variável completude através da utilização do teste t

	Valor de t	Graus de liberdade	Valor de p	Diferença média
Completude	1,554	16	0,140	12,5

Fonte: Autora (2021)

4.3.2 Avaliação do número de elementos mapeados

Para avaliar o número de elementos mapeados, foi proposta a questão de pesquisa QP2: “A utilização da abordagem Elicit4Safety tem efeito sobre a completude do mapeamento realizado em iStar4Safety?”. A métrica utilizada para calcular o número de elementos mapeados foi a soma dos seguintes elementos mapeados em iStar4Safety: objetivos de segurança, perigos, tarefas de segurança e recursos de segurança.

O Quadro 15 mostra os resultados da análise estatística descritiva do número de elementos mapeados dos dois tratamentos:

Quadro 15 - Análise Descritiva da Variável Número de Elementos Mapeados

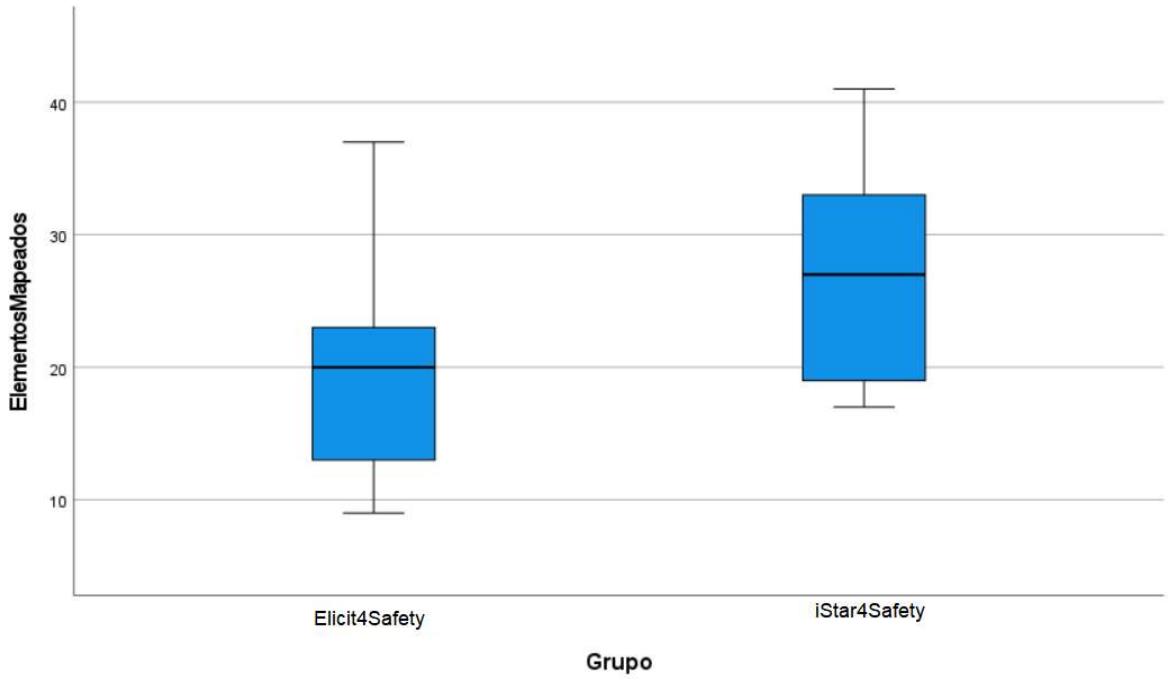
Número de Elementos Mapeados		
	Média	Mediana
Elicit4Safety	19,75	20
iStar4Safety	27,10	27

Fonte: Autora (2021)

A Figura 44 mostra o *boxplot* do número de elementos mapeados utilizando previamente a abordagem Elicit4Safety e quando se mapeado diretamente em iStar4Safety. A mediana, o primeiro quartil e o terceiro quartil mostram que os

participantes que realizaram a modelagem diretamente em iStar4Safety, sem a utilização prévia do Elicit4Safety conseguiram mapear mais elementos.

Figura 44 - Boxplot dos elementos mapeados



Fonte: Autora (2021)

O Quadro 16 mostra os resultados da análise estatística através da realização do teste t. A diferença média entre os tratamentos (-7,35) indica que o grupo experimental mapeou menos elementos do que o grupo de controle. No entanto, o valor de p obtido (t(16) = -1,757; p=0,098) indica que não existem diferenças estatísticas significativas entre os tratamentos. Sendo assim, não foi possível rejeitar a $H0_2$: *O número de elementos mapeados com a utilização da abordagem Elicit4Safety é igual aos gerados mapeando diretamente em iStar4Safety.*

Quadro 16 - Análise da variável número de elementos mapeados através da utilização do teste t

	Valor de t	Graus de liberdade	Valor de p	Diferença média
Elementos Mapeados	-1,757	16	0,098	-7,35

Fonte: Autora (2021)

4.3.3 Avaliação do tempo gasto para realizar a modelagem em iStar4Safety

Para avaliar o tempo gasto modelos, foi proposta a questão de pesquisa QP3: “A utilização da abordagem Elicit4Safety tem efeito sobre o tempo para realizar a modelagem em iStar4Safety?”. A métrica utilizada para calcular o tempo foi a quantidade de minutos que os participantes realizaram a atividade solicitada no experimento. Enquanto o grupo de controle lia a descrição do sistema robótico MIRAS e mapeava os elementos de segurança diretamente no iStar4Safety, o grupo experimental tinha duas atividades a serem realizadas: preencher o questionário utilizando a ferramenta Elicit4Safety, gerar um relatório e a partir deste relatório eles deveriam realizar manualmente a modelagem em iStar4Safety.

O Quadro 17 mostra os resultados da análise estatística descritiva do tempo para realizar a modelagem do sistema robótico MIRAS utilizando os dois tratamentos, de modo que seja visível que a média de tempo entre os dois tratamentos é bem parecida.

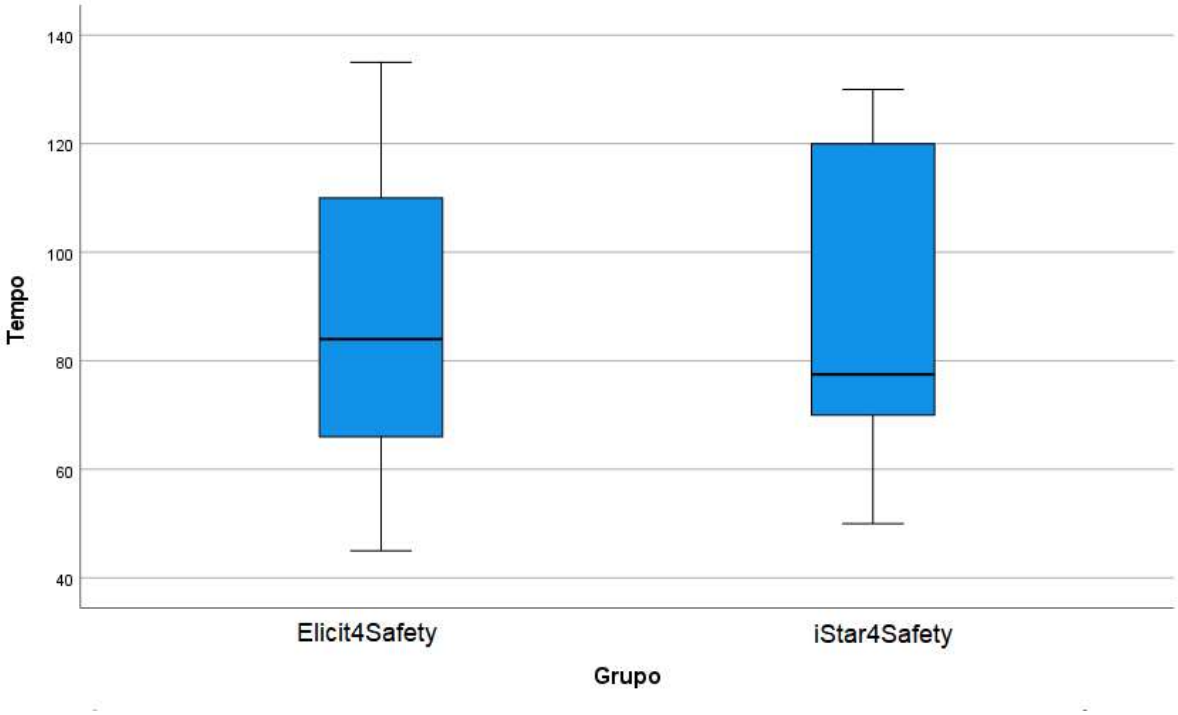
Quadro 17 - Análise Descritiva da Variável Tempo

Tempo em Minutos		
	Média	Mediana
Elicit4Safety	87,5	84
iStar4Safety	86,7	77,5

Fonte: Autora (2021)

A Figura 45 mostra o *boxplot* do tempo que os grupos levaram para finalizar a atividade e entregar o mapeamento do sistema robótico MIRAS em iStar4Safety. A mediana mostra que o tempo utilizando previamente a abordagem Elicit4Safety foi maior do que os que utilizaram apenas o iStar4Safety. Podemos afirmar que esta diferença no tempo se dá pois o grupo experimental possuía duas atividades a serem realizadas, enquanto o grupo de controle possuía apenas uma.

Figura 45 - Boxplot referente ao Tempo



Fonte: Autora (2021)

O Quadro 18 mostra os resultados da análise estatística através da realização do teste t. A diferença média entre os tratamentos (0,800) indica que o grupo experimental levou um pouco mais de tempo para realizar a atividade do que o grupo de controle. No entanto, de acordo com o valor de p obtido (t(16) = 0,057; p=0,955) indica que não existem diferenças estatísticas significativas entre os tratamentos. Sendo assim, não foi possível rejeitar a $H0_3$: *O tempo utilizado para mapear com auxílio do Elicit4Safety é igual ao utilizado para mapear diretamente em iStar4Safety.*

Quadro 18 - Análise da variável tempo através da utilização do teste t

	Valor de t	Graus de liberdade	Valor de p	Diferença média
Tempo	0,057	16	0,955	0,800

Fonte: Autora (2021)

4.3.4 Conclusões obtidas através da análise quantitativa

Os resultados destes experimentos não permitiram a rejeição de nenhuma hipótese nula (todos os valores de p são maiores que 0,05). Portanto, não podemos dizer que os resultados são estatisticamente significantes, exigindo sua replicação com amostras maiores. Para conseguir validar os resultados através dos testes estatísticos, será necessário a realização de novos experimentos, com mais participantes.

Contudo, para esta amostra, observamos que a média da completude do grupo que usou a abordagem Elicit4Safety foi superior à média do grupo que mapeou diretamente em iStar4Safety. Já a média do tempo gasto do grupo que usou a abordagem Elicit4Safety foi similar a média do grupo que mapeou diretamente em iStar4Safety. Enquanto a média do número de elementos mapeados do grupo que usou a abordagem Elicit4Safety foi inferior à média do grupo que mapeou diretamente em iStar4Safety.

É importante ressaltar que o grupo que usou abordagem Elicit4Safety tinha mais atividades a serem realizadas do que o grupo de controle pois eles deveriam inicialmente preencher as informações na ferramenta de acordo com a descrição do sistema, depois gerar um relatório e por fim realizar a modelagem em iStar4Safety. Já o grupo de controle precisava apenas ler a descrição do sistema e realizar o mapeamento diretamente em iStar4Safety.

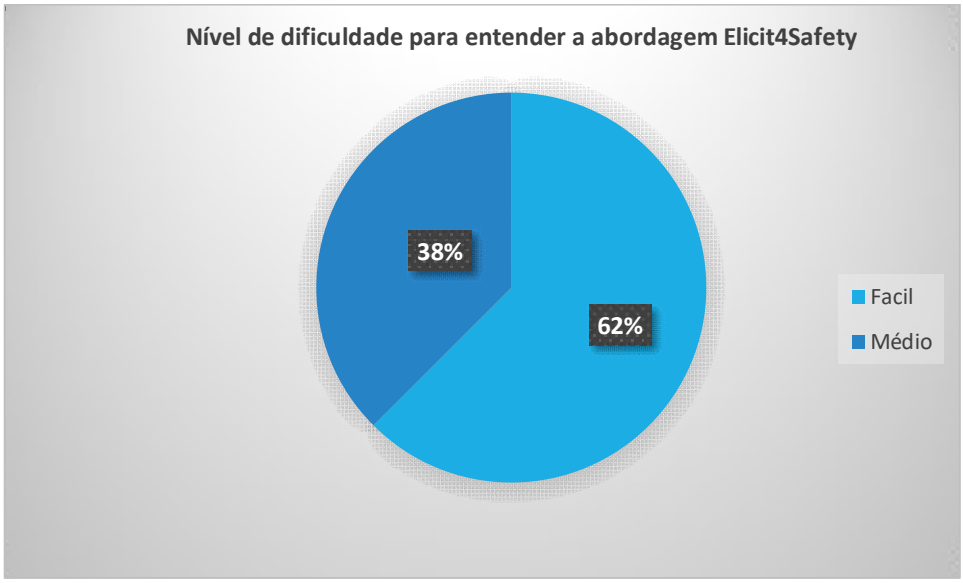
4.4 ANÁLISE QUALITATIVA DA ABORDAGEM ELICIT4SAFETY

A fim de fazer uma avaliação qualitativa da abordagem Elicit4Safety e do site que auxilia na utilização da abordagem, aplicamos um questionário para o grupo experimental (disponível no Anexo A.4) possuindo 13 questões. Algumas eram afirmações em que os participantes deveriam responder de acordo com a Escala Likert (LIKERT, 1932), onde 1 representa discordo totalmente e 5 representa concordo totalmente. Sendo elas:

4.4.1 Qual o nível de dificuldade em entender a abordagem Elicit4Safety?

A primeira pergunta realizada em nosso formulário, referia-se à dificuldade para compreender a abordagem Elicit4Safety, o gráfico da
Figura 46 representa os números obtidos e é possível perceber que a maioria dos participantes (5) classificou a abordagem como fácil de entender, enquanto três participantes informaram que a abordagem possui um nível médio de dificuldade para entender.

Figura 46 - Nível de dificuldade da abordagem

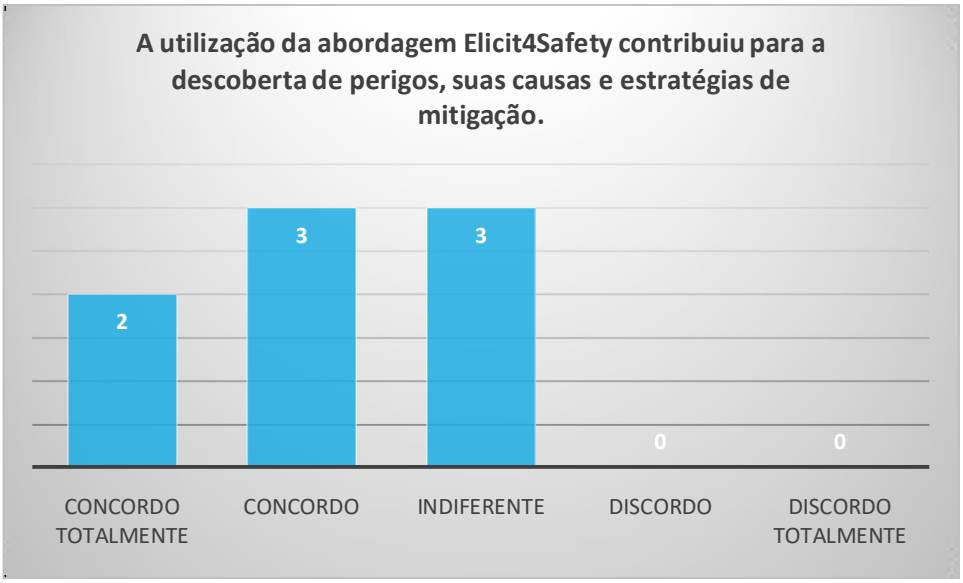


Fonte: Autora (2021)

4.4.2 A utilização da abordagem Elicit4Safety contribuiu para a descoberta de perigos, suas causas e estratégias de mitigação

Os participantes, em sua maioria, disseram que o uso da abordagem contribuiu para a modelagem de perigos, suas causas e estratégias de mitigação, conforme gráfico apresentado na Figura 47. Dois participantes concordaram totalmente, três concordaram e três foram indiferentes.

Figura 47 – Contribuição da abordagem Elicit4Safety

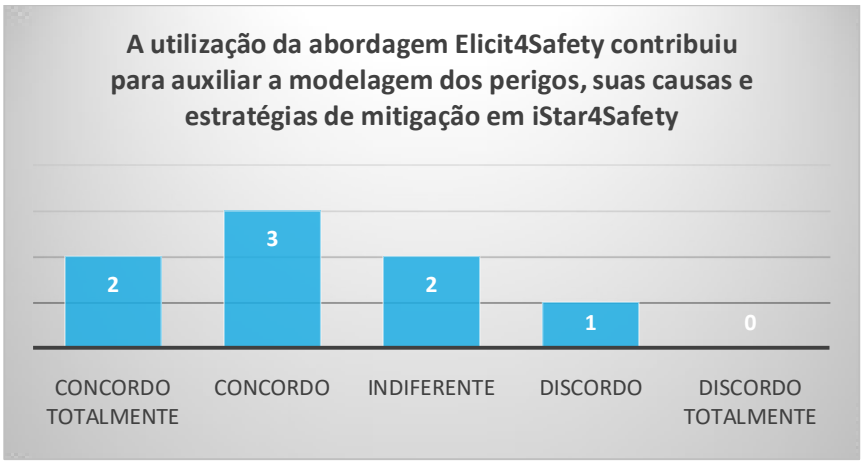


Fonte: Autora (2021)

4.4.3 A utilização da abordagem Elicit4Safety contribuiu para auxiliar a modelagem dos perigos, suas causas e estratégias de mitigação em iStar4Safety

O terceiro item do questionário foi se a utilização da abordagem contribuiu para auxiliar a modelagem de perigos, suas causas e estratégias de mitigação no iStar4Safety, nesta questão, cinco participantes concordaram, dois foram indiferentes e um discordou, conforme mostra a Figura 48.

Figura 48 - Contribuição para auxiliar na modelagem

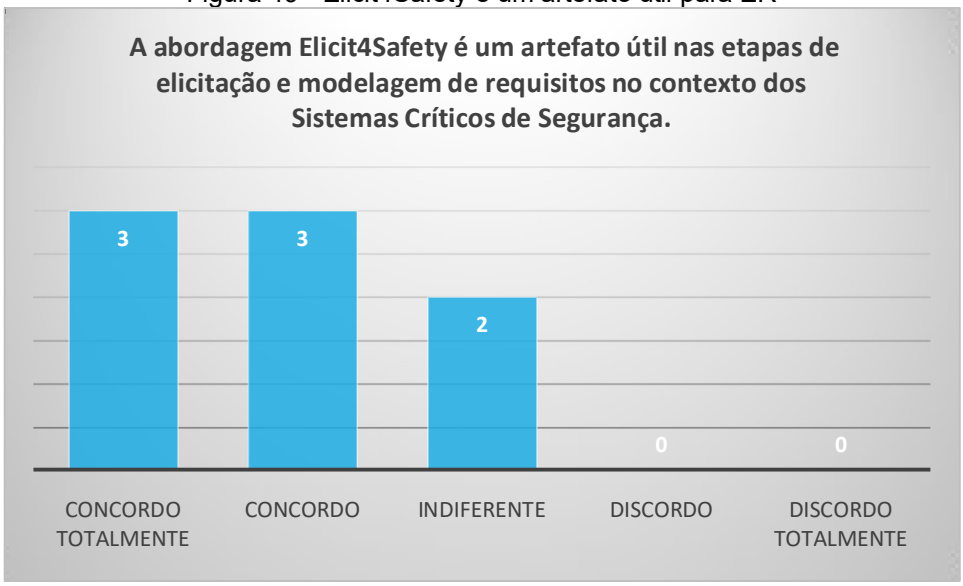


Fonte: Autora (2021)

4.4.4. A abordagem Elicit4Safety é um artefato útil nas etapas de elicitação e modelagem de requisitos no contexto dos Sistemas Críticos de Segurança

A Figura 49 mostra os resultados obtidos no item “A abordagem é um artefato útil nas etapas de elicitação e modelagem de requisitos no contexto dos Sistemas Críticos de Segurança”, conforme pode-se visualizar no gráfico, a maioria concordou que a abordagem é útil para a engenharia de requisitos em SCSs, apenas dois se posicionaram como indiferente.

Figura 49 - Elicit4Safety é um artefato útil para ER

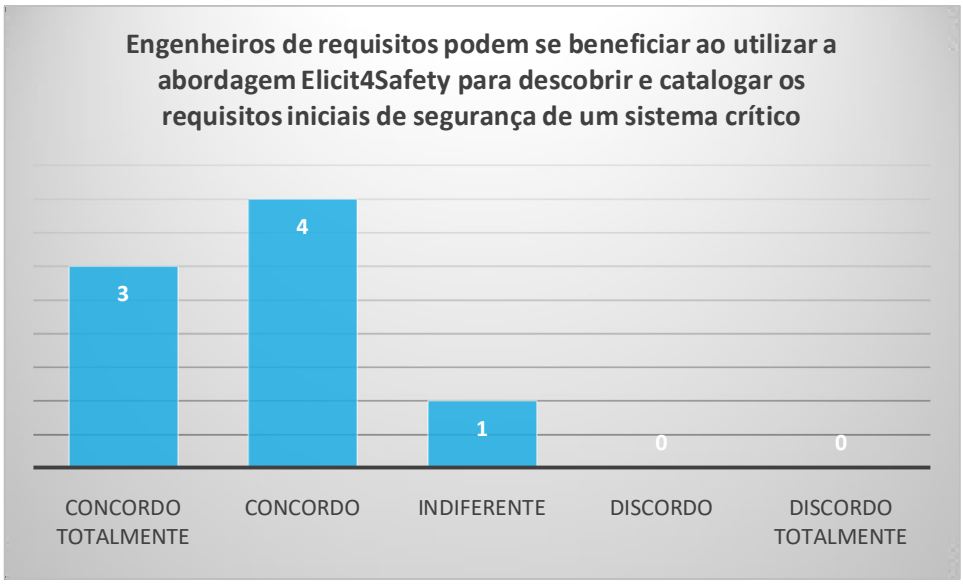


Fonte: Autora (2021)

4.4.5 Engenheiros de requisitos podem se beneficiar ao utilizar a abordagem Elicit4Safety para descobrir e catalogar os requisitos iniciais de segurança de um sistema crítico

A Figura 50 apresenta os resultados do item que trata do benefício da abordagem Elicit4Safety para que os engenheiros de requisitos descubram e cataloguem os requisitos iniciais de segurança de um SCS. Como pode-se observar no gráfico, a maioria concordou que os engenheiros de requisitos podem se beneficiar com a utilização da abordagem, apenas um ficou indiferente.

Figura 50 - Engenheiros de Requisitos podem se beneficiar da abordagem Elicit4Safety

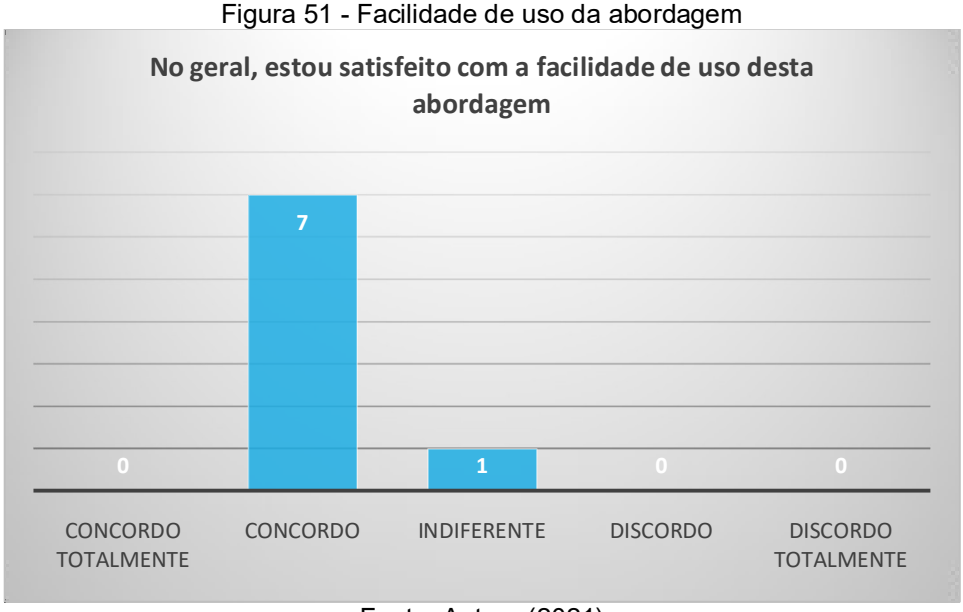


Fonte: Autora (2021)

4.4.6 Estou satisfeito com a facilidade de uso desta abordagem

A Figura 51 mostra o gráfico com as respostas obtidas na afirmação “No geral, estou satisfeito com a facilidade de uso desta abordagem” e como pode-se observar a maioria concordou, apenas um ficou indiferente.

Figura 51 - Facilidade de uso da abordagem

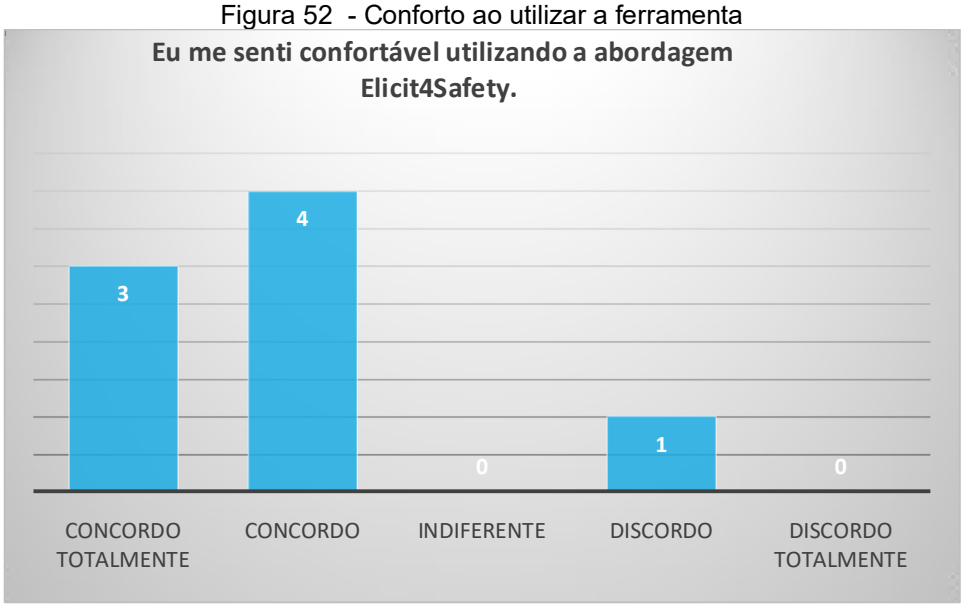


Fonte: Autora (2021)

4.4.7 Eu me senti confortável utilizando a abordagem Elicit4Safety.

No sentido de se sentir confortável utilizando a abordagem, três concordaram totalmente, quatro concordaram e apenas um discordou, como pode ser visualizado no gráfico da Figura 52.

Figura 52 - Conforto ao utilizar a ferramenta

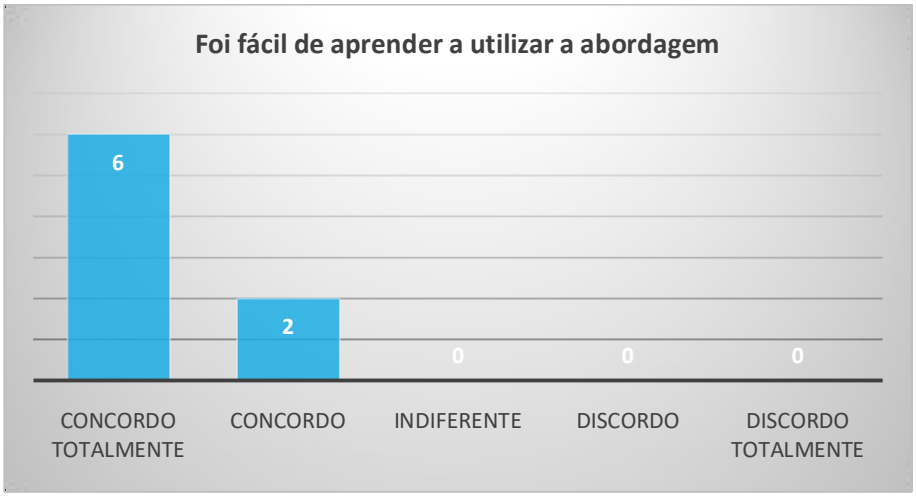


Fonte: Autora (2021)

4.4.8 Foi fácil de aprender a utilizar a abordagem

No quesito facilidade no aprendizado da abordagem, todos concordaram que é fácil de aprender, conforme pode ser visualizado na Figura 53.

Figura 53 - Facilidade no aprendizado da abordagem Elicit4Safety

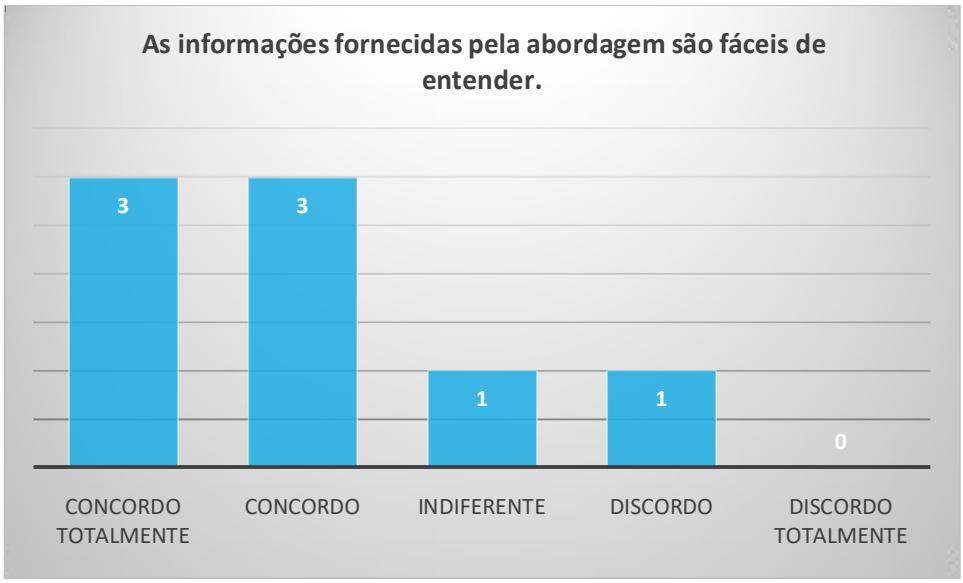


Fonte: Autora (2021)

4.4.9 As informações fornecidas pela abordagem são fáceis de entender.

Quanto às informações fornecidas pela abordagem, três concordaram totalmente que elas são fáceis de entender, três concordaram, um foi indiferente e um discordou, conforme mostra o gráfico da Figura 54.

Figura 54 - As informações fornecidas pela abordagem são fáceis de entender.

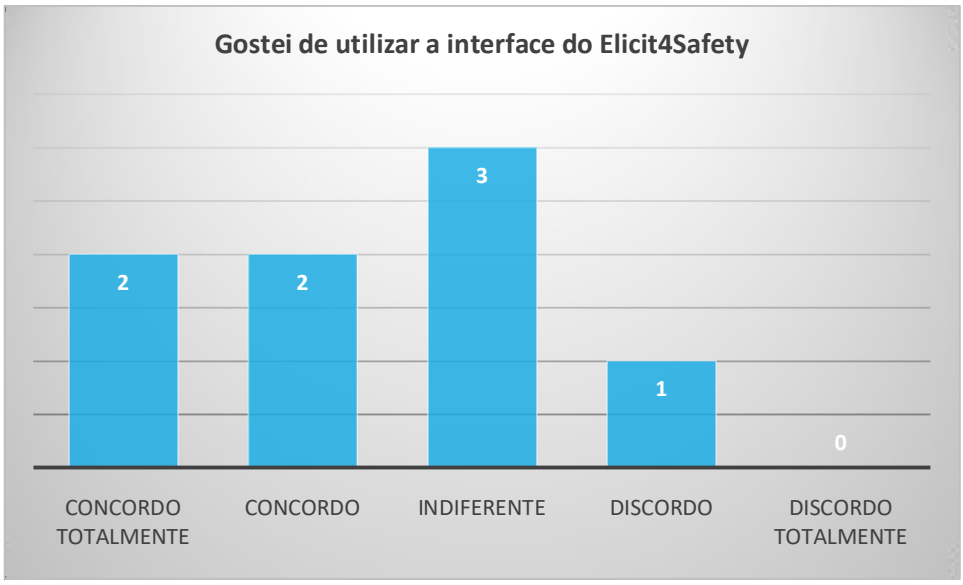


Fonte: Autora (2021)

4.4.10 Gostei de utilizar a interface do Elicit4Safety

Como pode ser visualizado no gráfico da Figura 55, a maioria concordou que gostou de utilizar a interface da ferramenta do Elicit4Safety, enquanto três ficaram indiferente e um participante discordou.

Figura 55 - Interface do sistema Elicit4Safety

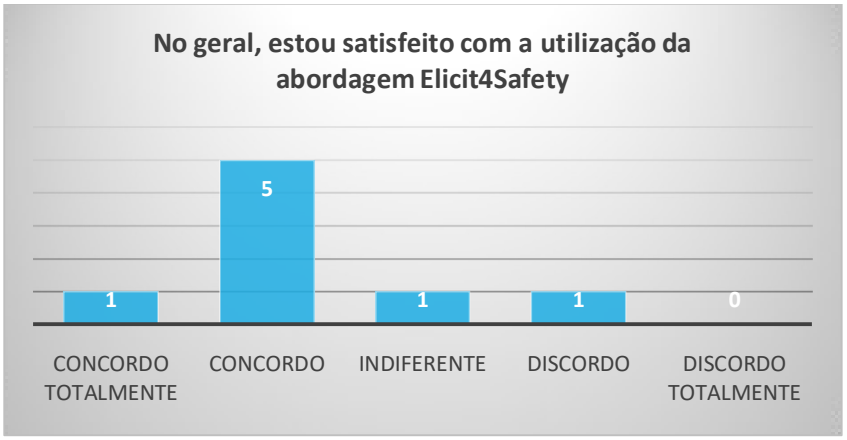


Fonte: Autora (2021)

4.4.11 No geral, estou satisfeito com a utilização da abordagem Elicit4Safety

A décima primeira afirmação do questionário, trata da satisfação na utilização da abordagem Elicit4Safety, um participante concordou totalmente que ficou satisfeito ao utilizar a abordagem, cinco concordaram, um foi indiferente e um discordou. Vale ressaltar que o participante que discordou desta afirmação, foi o mesmo que não gostou de utilizar a interface do sistema e que achou que as informações do sistema não são fáceis de entender. O gráfico com os resultados é apresentado na Figura 56.

Figura 56 - Satisfação na utilização da abordagem



Fonte: Autora (2021)

4.4.12 Para qual nível de profissionais você recomendaria o Elicit4Safety?

A décima segunda pergunta do questionário refere-se ao nível de profissionais que o Elicit4Safety deve ser recomendado. Seis participantes informaram que indicariam para engenheiros de software com experiência em *safety* e sem experiência em *safety*, um informou que indicava apenas para engenheiros de software com experiência em *safety* e um informou que não indicava para nenhum. O gráfico da Figura 57 apresenta os resultados.

Figura 57 - Profissionais que os participantes recomendariam a utilização do Elicit4Safety

Fonte: Autora (2021)

4.4.13 Sugestões para melhoria da ferramenta Elicit4Safety

Na décima terceira pergunta, nós perguntamos aos participantes se eles possuíam sugestões para melhoria da ferramenta Elicit4Safety e obtivemos as seguintes observações para aprimorar a ferramenta:

1. “Melhorar a usabilidade do sistema, não ficou claro em alguns pontos o que estava preenchendo e teve algumas horas que preenchi no local errado e só depois notei”.
2. "A técnica tem bom potencial, mas precisa de algumas mudanças de usabilidade, no meu caso particular, enfrentei provavelmente o pior caso de uso onde, por conta de um pequeno erro de digitação tive que reiniciar o processo todo do começo e refazer todo o modelo. A visão em formato de

treeview vertical também atrapalha quando o número de entidades começa a aumentar, fica muito confuso a navegação de onde cada coisa começa e termina. O relatório gerado é bom, mas também sofre do problema de dificultar a leitura, pois não se sabe onde começa uma entidade e onde termina. Sei da possibilidade de retrair as entidades, porém, num momento de revisão do processo é inevitável ter todas as entidades em formato expandido. A possibilidade de voltar e corrigir o modelo gerado sem ter que recomeçar o processo todo do zero é uma melhoria essencial. Um indicativo de cor pode melhorar a usabilidade. Uma forma super simples de fazer isso é utilizar uma cor padrão do sistema e tons cada vez mais claros ou mais escuros dessa cor a medida que a arvore vai aumentando, essa cor ficaria no fundo de cada div. Por utilizar uma cor em tons diferentes, essa técnica também é inclusiva para quem possui daltonismo."

3. “Talvez destacar um pouco partes do relatório tornaria mais rápida a interpretação dele após estar pronto (separação por safetygoals, por ex).”
4. “A ordem dos itens gerados no relatório poderia ser mais organizada. Ex: agrupar os Hazard. Talvez colorir os *textfields* dos formulários de acordo com o tipo do elemento (Hazard, SafetyGoal, SafetyTask, etc) IStarSafety relacionado ajudaria mais em diferenciar cada um durante a revisão”.
5. “Seria muito útil poder retornar e editar os campos, após gerar o relatório em PDF. Identifiquei alguns ajustes necessários e precisei refazer tudo”.
6. “Análise: 1. Quando formamos a cadeia no Elicit4safety, há informações que não serão utilizadas, estas "confundem" na hora de inserir na iStar4Safety. Poderia sinalizar com cores aquelas informações que não serão usadas no modelo iStar4Safety; 2. Na impressão, a organização em tópicos deveria obedecer a uma hierarquia e/ou tabulação, ou seja, objetivos alinhados a objetivos, tarefas com tarefas, perigos com perigos e sucessivamente. Demorei mais para achar os "alinhamentos" hierárquicos do que a informação propriamente dita; 3. Na operação do Elicit4Safety, eu confundi a análise entre os atores, assim, tive que refazer. Mas, quando inseri as informações no segundo ator, parte dos dados do primeiro apareceu "pré-preenchido"”.
7. “Depois de finalizar poder voltar e editar algo que escreveu errado”.

4.4.14 Conclusões obtidas através da análise qualitativa

De acordo com as respostas dos participantes no questionário e os dados qualitativos obtidos, é possível concluir que a abordagem é fácil de entender e é fácil de ser utilizada. Além do mais, a maioria dos participantes responderam que ela contribui para a descoberta e modelagem de perigos, assim como é um artefato útil nas etapas de elicitação e modelagem de requisitos, de modo que possa beneficiar os engenheiros de requisitos. Foi possível verificar que a maioria dos participantes estão satisfeitos com a abordagem de modo geral, no entanto, a questão da interface foi um ponto bem discutido por eles nas sugestões de alteração.

Sendo assim, como trabalho futuro dessa dissertação, deverá ser elaborada uma nova versão da ferramenta com uma interface mais convidativa e que permita que os usuários possam voltar e corrigir os dados previamente digitados. Ademais, serão estudadas outras propostas de melhoria para serem aplicadas.

4.5 AMEAÇAS À VALIDADE

De acordo com Travassos *et al.* (2012) “a questão fundamental a respeito dos resultados do experimento é quão válidos são eles”, sendo assim, entende-se que é de extrema importância que tenhamos resultados que sejam considerados válidos. Portanto, nesta subseção serão apresentadas as ameaças ligadas à validade do quasi-experimento proposto.

É importante salientar que diante da situação pandêmica em que nos encontramos, foi necessário que realizássemos as atividades do quasi-experimento de maneira remota, não sendo possível administrar plenamente o ambiente controlado. No entanto, tentamos diminuir esta ameaça pedindo para que os alunos estivessem logados na plataforma Google Meet enquanto realizavam a atividade.

4.5.1 Validade de conclusão

Segundo Wohlin (2012) as ameaças à validade de conclusão referem-se a questões que afetam a habilidade de tirar a conclusão certa acerca do

relacionamento entre o tratamento e o resultado de um experimento. Foram identificadas as seguintes ameaças a validade de conclusão neste trabalho: (1) Baixo poder estatístico: esta ameaça se deu ao baixo número de participantes no experimento. (2) Experiência dos sujeitos: os participantes do experimento são iniciantes na área de *safety*. Essas ameaças limitam a possibilidade de generalização dos resultados obtidos, deste modo, os resultados deste estudo são considerados preliminares e não podem ser generalizados.

4.5.2 Validade interna

De acordo com Wohlin (2012) as ameaças à validade interna são influências que podem afetar a variável independente no que diz respeito à causalidade, sem o conhecimento do pesquisador. Deste modo, eles ameaçam a conclusão sobre uma possível relação causal entre o tratamento e o resultado. A fim de reduzir o viés de seleção, foi realizada uma distribuição aleatória dos participantes nos grupos experimental e de controle. Além disso, tentamos minimizar a ameaça de difusão ou imitação de tratamentos, uma vez que só apresentamos a abordagem Elicit4Safety ao grupo experimental.

4.5.3 Validade de construto

Para Wohlin (2012) a validade de construto trata da generalização do resultado do experimento para a teoria que está por trás do mesmo. Encontramos a seguinte ameaça a validade deste tipo neste trabalho: viés mono-operação – esta ameaça ocorre quando há apenas uma variável independente. Quanto às ameaças de receio de avaliação e adivinhação de hipóteses, tentamos mitigar da seguinte maneira:

- Receio de avaliação: esta ameaça foi mitigada uma vez que os participantes do quasi-experimento foram voluntários.
- Adivinhação de hipóteses: esta ameaça foi mitigada uma vez que os sujeitos não foram informados sobre o que seria investigado.

Além do mais, como os sujeitos não tinham experiência anterior em

Segurança antes do curso e todos receberam o mesmo treinamento, bem como materiais de apoio para auxiliar na modelagem dos requisitos de segurança e a descrição do sistema que seria modelado. Portanto, podemos concluir que as diferenças nos resultados estão relacionadas à técnica que utilizaram (Elicit4Safety ou não).

4.5.4 Validade externa

De acordo com Wohlin (2012) as ameaças à validade externa limitam a capacidade de generalizar os resultados do experimento para a prática industrial. Uma ameaça deste tipo detectada neste estudo foi o fato de os participantes serem alunos de graduação e iniciantes na área de *safety* e elicitação de requisitos, no entanto, de acordo com Svahnberg *et al.* (2008) foi demonstrado que não há necessariamente muita diferença entre alunos e profissionais em muitos ambientes experimentais.

Outra ameaça deste tipo foi que nós não conseguimos reproduzir um ambiente totalmente controlado por conta da pandemia do Coronavirus, que impediu que o quasi-experimento fosse realizado de modo presencial. Como o quasi-experimento precisou ser realizado online, através do Google Meet, nós tentamos reproduzir um ambiente controlado pedindo que os alunos ficassem na sala virtual enquanto realizavam as atividades.

4.6 CONCLUSÃO DO CAPÍTULO

Este capítulo apresentou os resultados obtidos através da aplicação do quase experimento e foi dividido entre os dados quantitativos e qualitativos. Infelizmente, não conseguimos rejeitar nenhuma hipótese nula, uma vez que o valor de p não foi menor que 0.05, indicando que os dados não possuem diferença significativa. Porém, a partir da análise estatística descritiva por meio dos *boxplot* conseguimos visualizar algumas diferenças entre os tratamentos. Vale ressaltar que a situação em que estamos vivendo, de pandemia, dificultou um pouco a realização desse quasi-experimento, não permitindo criar um ambiente totalmente controlado e garantir que os alunos estivessem compreendendo e realizando as atividades da

maneira correta.

Quanto aos dados qualitativos, foi possível observar que a abordagem é fácil de entender e utilizar.

5 CONCLUSÃO

Neste capítulo discutimos o que foi realizado, apresentamos as limitações encontradas, relatamos as contribuições obtidas e propomos os trabalhos futuros.

5.1 DISCUSSÃO

O principal objetivo do presente trabalho foi a criação de uma abordagem que permitisse auxiliar a integração da engenharia de requisitos e de segurança através do processo de levantamento e modelagem de requisitos iniciais de segurança do domínio dos SCSs. A abordagem proposta consiste em um conjunto de perguntas que visam auxiliar na descoberta de requisitos de segurança e mapeamento dos perigos oriundos do sistema, e pode ser utilizada tanto quanto questionário, quanto como roteiro para uma entrevista estruturada. Ademais, a abordagem Elicit4Safety está relacionada com a notação para modelagem de requisitos de segurança denominada iStar4Safety, sendo assim, recomenda-se o mapeamento dos requisitos nesta notação após a descoberta dos mesmos.

A fim de atingir o principal objetivo, foram definidos alguns objetivos específicos, sendo eles:

(1) Realizar um levantamento do estado da arte sobre a elicitação de requisitos para SCSs, para identificar as técnicas existentes e suas limitações. Este objetivo foi satisfeito uma vez que fizemos uma busca na literatura, através das seguintes bibliotecas digitais: ACM Digital Library, IEEE Xplore Digital Library, ScienceDirect e Springer, a fim de localizar artigos e livros relevantes para a construção da dissertação. Através deste levantamento bibliográfico, nós localizamos algumas técnicas de elicitação para sistemas críticos de segurança: Du *et al.* (2014), Martins e de Oliveira (2014) e Provenzano *et al.* (2017).

(2) Desenvolver uma abordagem que facilite a elicitação de requisitos iniciais de segurança no domínio dos SCSs. Atingimos este objetivo ao desenvolver a abordagem Elicit4Safety, que tem o intuito de ser integrada com a modelagem de requisitos de acordo com a notação iStar4Safety. A abordagem construída se trata de um conjunto de perguntas que integram os conceitos vistos na notação e na Análise Preliminar de Perigos.

(3) Desenvolvimento de uma ferramenta para dar suporte a abordagem. Atingimos este objetivo ao propor uma ferramenta web, utilizando as tecnologias

CSS, HTML e JavaScript para dar suporte às perguntas, uma vez que algumas delas seguem uma estrutura de árvore. A ferramenta pode ser acessada através do seguinte link: https://cin.ufpe.br/~sdm2/Elicit4Safety/.

(4) Atrelar a abordagem desenvolvida com a técnica de modelagem de requisitos iniciais de segurança denominada iStar4Safety. Atingimos este objetivo específico ao criar diretrizes para utilização da ferramenta e conversão das informações obtidas no relatório da ferramenta para a modelagem iStar4Safety, tendo em vista que o mapeamento ainda não é realizado de maneira automatizada.

(5) Avaliar a abordagem proposta a partir da realização de um quasi-experimento. Para atingir este objetivo específico, realizamos um quasi-experimento com alunos da graduação e pós-graduação das disciplinas de Engenharia de Requisitos e Especificação e Validação de Sistemas do Centro de Informática da Universidade Federal de Pernambuco e a partir deste quasi-experimento extraímos dados quantitativos e qualitativos, estes últimos foram obtidos através do preenchimento de um questionário pelo grupo experimental pós utilização da ferramenta. Na avaliação quantitativa, buscamos analisar três variáveis dependentes: completude, tempo e número de elementos mapeados, e, infelizmente, por conta do baixo número de amostras, não conseguimos atingir o nosso objetivo de rejeitar as hipóteses nulas, mas a análise estatística descritiva através dos *boxplots* nos mostrou algumas diferenças entre os tratamentos. Quanto aos dados qualitativos, foi possível observar que os participantes em sua maioria classificam a abordagem como fácil de ser entendida e utilizada. No entanto, a interface do sistema é um ponto que deverá ser tratado no futuro.

5.2 LIMITAÇÕES

Atualmente o mundo vem passando por um período bem difícil, caracterizado pela pandemia do Coronavírus, que assim como afetou o modo de trabalho e a vida de muitas pessoas, também acabou afetando os resultados da presente pesquisa. Sendo assim, foram percebidas algumas limitações durante o desenvolvimento desta dissertação.

A principal limitação ocorrida devido à pandemia, foi referente ao quasi-experimento realizado para avaliação da abordagem Elicit4Safety, que precisou

ocorrer de forma *online*, através de reuniões pela plataforma Google Meet e atividades realizadas por meio do Google Sala de Aula. Portanto, não foi possível criar um ambiente totalmente controlado. Tentamos minimizar esta ameaça solicitando para que os alunos começassem a realizar o exercício no horário da aula, em uma chamada de Google Meet, mas permitindo que, caso necessário, eles estendessem a realização da atividade para após o horário da aula, tendo em vista que alguns tiveram problema de acesso ao computador ou dificuldades de conexão no horário da aula. As ameaças à validade do estudo estão representadas na seção 4.6.

Outra limitação do presente trabalho ocorrida por conta da pandemia do Coronavírus foi a impossibilidade de realizar o experimento em um projeto real, abrangendo profissionais da área de engenharia de segurança e engenharia de requisitos.

5.3 CONTRIBUIÇÕES

Esta dissertação apresentou o desenvolvimento da abordagem Elicit4Safety que permite auxiliar na descoberta dos requisitos iniciais de segurança de um SCSs, bem como a integração com a técnica de modelagem de requisitos iStar4Safety e a técnica de análise preliminar de perigos (PHA). Outras contribuições deste trabalho são:

• Disponibilização de uma ferramenta de mesmo nome – Elicit4Safety – para auxiliar no processo de descoberta de requisitos funcionais de segurança e outros elementos ligados ao iStar4Safety e à técnica PHA, bem como no processo de aprendizagem do iStar4Safety;

• Um guia com diretrizes para utilização da abordagem Elicit4Safety; e

• Modelagem do MIRAS Robot em iStar4Safety.

5.4 TRABALHOS FUTUROS

Alguns trabalhos devem ser realizados para complementar ou dar prosseguimento à proposta desta dissertação, tais como:

• Fazer uma nova versão da ferramenta Elicit4Safety, abrangendo as sugestões

fornecidas pelos alunos na avaliação da ferramenta;

• Gerar automaticamente o modelo em iStar4Safety a partir do relatório obtido na ferramenta Elicit4Safety;

• Realizar novos experimentos a fim de analisar melhor a utilização da abordagem Elicit4Safety e da nova ferramenta;

• Geração automática de planilhas de Análise Preliminar de Perigos;

• Comparar a abordagem com outras técnicas para elicitação de requisitos de sistemas críticos através de experimentos;

• Avaliar a abordagem com engenheiros de requisitos que trabalham no domínio de Sistemas Críticos de Segurança;

• Avaliar se a utilização o Elicit4Safety beneficia outras abordagens para modelagem de requisitos de segurança; e

• Atrelar a abordagem Elicit4Safety com normas regulatórias.

REFERÊNCIAS

BATISTA, Edinelson Aparecido. Uma taxonomia facetada para tecnicas de elicitação de requisitos: Edinelson Aparecido Batista. 2003. 150p. Dissertação (mestrado) - Universidade Estadual de Campinas, Instituto de Computação, Campinas, SP. Disponível em: <http://www.repositorio.unicamp.br/handle/REPOSIP/276373>. Acesso em: 3 ago. 2018.

BROOMFIELD, E. J.; CHUNG, P. W. H. Safety assessment and the software requirements specification. Reliability Engineering & System Safety, v. 55, n. 3, p. 295-309, 1997.

DU, Junwei; WANG, Jiqiang; FENG, Xiaogang. A safety requirement elicitation technique of safety-critical system based on scenario. In: International Conference on Intelligent Computing. Springer, Cham, 2014. p. 127-136.

DUARTE, Felipe Lima et al. Uma abordagem orientada a modelos e para engenharia de requisitos de sistemas de sistemas. 2018.

EASTERBROOK, S.; SINGER, J.; STOREY, M.A.; DAMIAN, D. Selecting empirical methods for software engineering research. In: Guide to advanced empirical software engineering. Springer, London, 2008. p. 285-311.

ERICSON, Clifton A. Hazard analysis techniques for system safety. John Wiley & Sons, 2015.

FIRESMITH, Donald G. A taxonomy of safety-related requirements. Requirements Engineering, 2004.

FONSECA, Liliane Sheyla da Silva. An instrument for reviewing the completeness of experimental plans for controlled experiments using human subjects in software engineering. Tese (Doutorado em Ciência da Computação) – Universidade Federal de Pernambuco, Recife, 2016.

GIL, A. C. Como Elaborar Projetos de Pesquisa. 5.ed. São Paulo: Atlas, 2010

GOWEN, Lon D.; COLLOFELLO, James S.; CALLISS, Frank W. Preliminary hazard analysis for safety-critical software systems. In: Eleventh Annual International Phoenix Conference on Computers and Communication [1992 Conference Proceedings]. IEEE, 1992. p. 501-508.

GRANT, Emanuel S. Requirements engineering for safety critical systems: An approach for avionic systems. In: 2016 2nd IEEE International Conference on Computer and Communications (ICCC). IEEE, 2016. p. 991-995. Avionic Systems (2016).

IBM Corp. Released 2020. IBM SPSS Statistics for Windows, Version 27.0. Armonk, NY: IBM Corp.

IEEE100, “The Authoritative Dictionary of IEEE Standard Terms”, IEEE Press, 2000.

IGNÁCIO, Rafael Crispim; BENITTI, Fabiane. Improving the selection of requirements elicitation techniques: a faceted guide.

JURISTO, Natalia; MORENO, Ana M. Basics of software engineering experimentation. Springer Science & Business Media, 2013.

KAUSAR, S.; TARIQ, S; RIAZ, S.; KHANUM, A. Guidelines for the selection of elicitation techniques. In: 2010 6th International Conference on Emerging Technologies (ICET). IEEE, 2010. p. 265-269.

KNIGHT, John C. Safety critical systems: challenges and directions. In: Proceedings of the 24th international conference on software engineering. ACM, 2002. p. 547-550.
KOTONYA, Gerald; SOMMERVILLE, Ian. Requirements engineering: processes and techniques. Wiley Publishing, 1998.

KUMAR, S. Phani; RAMAIAH, P. Seetha; KHANAA, V. A methodology for building safer software based critical computing systems. In: 2010 IEEE 2nd International Advance Computing Conference (IACC). IEEE, 2010. p. 422-429.

LEVENE, H. Robust Tests for the equality of variance. In: Olkin, I(Ed.) Contributions to Probability and Statistics, Palo Alto, California: Stanford University Press, 1960. p.278– 292.

LEVESON, Nancy G. Safeware: system safety and computers. Addison-Wesley, 1995.

LEVESON, Nancy. Engineering a safer world: Systems thinking applied to safety. MIT press, 2011.

LIKERT, Rensis. A technique for the measurement of attitudes. **Archives of psychology**, 1932.

LUTZ, Robyn R. Software engineering for safety: a roadmap. In: Proceedings of the Conference on the Future of Software Engineering. 2000. p. 213-226.
MARÔCO, João. Análise Estatística com o SPSS Statistics.: 7ª edição. ReportNumber, Lda, 2018.

MARTINS, Luiz Eduardo Galvão; DE OLIVEIRA, Tiago. A case study using a protocol to derive safety functional requirements from fault tree analysis. In: 2014 IEEE 22nd International Requirements Engineering Conference (RE). IEEE, 2014. p. 412-419.

MARTINS, Luiz Eduardo G.; GORSCHEK, Tony. Requirements engineering for safety-critical systems: A systematic literature review. Information and software technology, v. 75, p. 71-89, 2016.

MEDIKONDA, Ben Swarup; PANCHUMARTHY, Seetha Ramaiah. A framework for software safety in safety-critical systems. ACM SIGSOFT Software Engineering Notes, v. 34, n. 2, p. 1-9, 2009.

MIL-STD-882E System Safety. [S.l.], 2012. Disponível em: <http://www.everyspec.com/MIL-STD/MIL-STD-0800-0899/MIL_STD_882D_934/>.

MIRAS, “Multimodal Interactive Robot for Assistance in Strolling,” Project supported by the French ANR (National Research Agency) under the TecSan (Healthcare Technologies) Program (ANR-08-TECS-009-04)

POHL, Klaus. Requirements engineering: fundamentals, principles, and techniques. Springer Publishing Company, Incorporated, 2010.

PROVENZANO, L.; HANNINE, K; ZHOU, J.; LUNDQVIST, K. An ontological approach to elicit safety requirements. In: 2017 24th Asia-Pacific Software Engineering Conference (APSEC). IEEE, 2017. p. 713-718.

RAGHAVAN, Sridhar; ZALENISK, Gregory; FORD, Gary. Lecture Notes on Requírements Elícítatíon. Educational Material CMU/SEI-94-EM-10. Software Engineering Institute, Carnigie Mellon University, Pittsburg, USA, 1994.

RIBEIRO, Moniky; CASTRO, Jaelson, VILELA, Jessyka, PIMENTEL, João. iStar4Safety: Uma Extensão de iStar para Modelagem de Requisitos de Segurança em Sistemas Críticos. In: **WER**. 2019ª.

RIBEIRO, Sarah Moniky Silva. Desenvolvimento de uma extensão da linguagem de modelagem iStar para sistemas críticos de segurança – iStar4Safety. 2019. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Pernambuco, Recife, 2019b.

SHAPIRO, S. S.; WILK, M. B. An analysis of variance test for normality (complete sample). Biometrika, Great Britain, v. 52, n. 3, p. 591-611, 1965.

SVAHNBERG, Mikael; AURUM, Aybüke; WOHLIN, Claes. Using students as subjects-an empirical evaluation. In: **Proceedings of the Second ACM-IEEE international symposium on Empirical software engineering and measurement**. 2008. p. 288-290.

SOMMERVILLE, Ian. Engenharia de Software. Tradução Ivan Bosnic e Kalinka G. de O. Gonçalves; revisão técnica Kechi Hirama. — 9. ed. — São Paulo: Pearson Prentice Hall, 2011.

STAKE, R. E. Pesquisa qualitativa: estudando como as coisas funcionam. Porto Alegre: Penso, 2011.

TORO, Amador D; JIMÉNEZ, Beatriz B. Metodología para la Elicitación de Requisitos de Sistemas de Software. Informe Técnico LSI-2000-10. Facultad de

Informática y Estadística Universidad de Sevilla, outubro, 2000.

TRAVASSOS, Guilherme Horta; GUROV, Dmytro; AMARAL, E. A. G. G. Introdução à engenharia de software experimental, Programa de Engenharia de Sistemas e Computação, COPPE/UFRJ, Relatório Técnico. 2002.

TSUMAKI, Toshihiko; TAMAI, Tetsuo. Framework for matching requirements elicitation techniques to project characteristics. Software Process: Improvement and Practice, v. 11, n. 5, p. 505-519, 2006.

VILELA, J.; CASTRO, J.; MARTINS, L.E.G; GORSCHEK, T. Integration between requirements engineering and safety analysis: A systematic literature review. Journal of Systems and Software, v. 125, p. 68-92, 2017a.

VILELA, Jéssyka et al. Specifying safety requirements with gore languages. In: **Proceedings of the 31st Brazilian Symposium on Software Engineering**. 2017b. p. 154-163.

VILELA, J.; CASTRO, J.; MARTINS, L.E.G; GORSCHEK, T. Safe-RE: a safety requirements metamodel based on industry safety standards. In: Proceedings of the XXXII Brazilian Symposium on Software Engineering. ACM, 2018a. p. 196-201.

VILELA, Jéssyka Flavyanne Ferreira. Uni-REPM SCS: a safety maturity model for requirements engineering process. 2018b.

VILELA, J.; CASTRO, J.; MARTINS, L.E.G; GORSCHEK, T. Safety Practices in Requirements Engineering: The Uni-REPM Safety Module. IEEE Transactions on Software Engineering, v. 46, n. 3, p. 222-250, 2020.

WILKINSON, Philip; MAVIN, Alistair. The Early Discovery of Requirements for Safety Critical Systems. In: Advances in Risk and Reliability Technology Symposium (ARRTS), Loughborough. 2015.

WOHLIN, C.; RUNESON, P.; HOST, M.; OHLSSON, M. C.; REGNELL, B. Experimentation in software engineering. Springer Science & Business Media, 2012.

YEOW, Eileen; KIA CHIAM, Yin. Integration of safety risk assessment techniques into requirement elicitation. Frontiers in Artificial Intelligence and Applications, [S. l.], v. 265, p. 256–270, 2014. Disponível em: https://doi.org/10.3233/978-1-61499-434-3-256

YOUSUF, Masooma; ASGER, M. Comparison of various requirements elicitation techniques. International Journal of Computer Applications, v. 116, n. 4, 2015.

YU, E. Modelling strategic relationships for process reengineering. PhD thesis, Department of Computer Science, University of Toronto, 1995. Also Technical Report DKBS-TR-94-6

APÊNDICE A – MATERIAL DE SUPORTE DO EXPERIMENTO

A.1 DESCRIÇÃO DO SISTEMA ROBÓTICO MIRAS A SER MAPEADO UTILIZANDO O ELICIT4SAFETY + ISTAR4SAFETY

1. Descrição do sistema

MIRAS (MIRAS, 2009) é um robô assistivo que tem como objetivo **auxiliar pessoas idosas a se levantarem, passearem e sentarem** quando não houver equipe médica disponível. O robô consiste em uma **base de rodas e duas alças que simulam um guidão móvel e possui alguns sensores** com o objetivo de garantir o correto funcionamento do sistema, sendo eles: (1) **sensor de força nas alças**, que busca garantir a estabilização da postura dos pacientes durante o levantamento, locomoção e navegação do paciente, (2) **sensor de colisão**, que permite que o robô caminhe sem colidir com objetos (3) s**ensor para detecção de posição de assento**, que permite que o robô identifique onde está o assento que o paciente irá se sentar.

Para que o paciente seja levantado corretamente, é necessário que sejam inseridos os **dados do paciente**, indicando a **altura** que as alças do robô devem estar para que o paciente se apoie corretamente. Quanto à caminhada, o robô está apto para desenvolver suas atividades **em ambientes confinados e previamente mapeados**, de modo que seja permitido identificar todos os obstáculos do trajeto que o robô irá circular com o paciente. Durante a caminhada com robô é necessário que o **sensor de colisão esteja funcionando perfeitamente**, a fim de evitar que o robô colida com algum objeto ou pessoa. Ademais, em caso de anormalidades, o robô deve soar um alarme de emergência.

O robô foi projetado para ser capaz de se mover de forma autônoma e navegar para os pacientes quando é chamado.

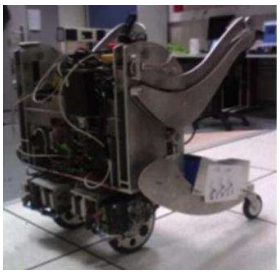
O presente sistema robótico possui alguns benefícios, como:

- Tornar a caminhada independente para pacientes com distúrbios de equilíbrio e orientação;
- Melhorar o condicionamento físico do idoso através da caminhada;
- Liberar a equipe médica para atos mais técnicos.

Figura 58 - Andador Clássico

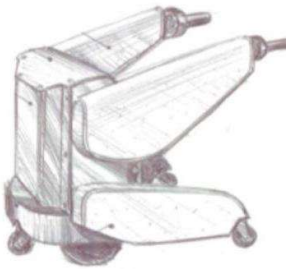
Fonte: Miras (2009)

Figura 59 - Robô Miras Experimental



Fonte: Miras (2009)

Figura 60 - Design do Robô



Fonte: Miras (2009)

Figura 61 - Protótipo do robô

Fonte: Miras (2009)

2. Funcionalidades do sistema

Dentre as funcionalidades do sistema, podemos destacar as seguintes:

• **Operação para caminhar com o paciente:** Esta função refere-se ao robô levar o paciente para uma caminhada. Existem algumas pré-condições para realização desta função: As alças devem estar na altura correta do paciente, as baterias devem estar suficientemente carregadas para ir e voltar ao ponto mais distante, além de gerar alarmes no caso de alguma emergência.

• **Operação para levantar o paciente:** Nesta função, o robô deverá levantar o paciente. Existem algumas pré-condições para realização desta função: O paciente deve estar sentado, o robô deve estar esperando para levantar o paciente, as baterias devem estar suficientemente carregadas para levantar e posteriormente

ajudar o paciente a se sentar, o robô deve estar na frente do paciente. A função possui as seguintes pós-condições: o paciente fica em pé, o robô fica no modo de caminhada.

• **Operação para sentar o paciente:** Nesta função, o robô deverá sentar o paciente. Existem algumas pré-condições para realização desta função: O paciente está em pé, o robô está esperando para sentar o paciente e é necessário a presença de um assento. A função possui as seguintes pós-condições: o paciente está sentado, o robô está no modo de espera.

• **Gerenciamento de perda de equilíbrio:** Esta função permite que o robô perceba quando o paciente perdeu o equilíbrio, através dos sensores instalados, e auxilie o paciente a retomar o equilíbrio.

• **Gerenciamento de alarmes:** Esta função permite que o robô acione alarmes quando detectar alguma anormalidade.

• **Chamada e movimento autônomo do robô:** Esta função gerencia o movimento autônomo dos robôs sem intervenção humana. Possui as seguintes pré-condições: as baterias estão suficientemente carregadas para realizar esta tarefa, o caminho até o paciente não deve possuir obstáculos. A pós-condição: o robô está no modo de espera.

Detecção de fim de uso e movimento para a posição de espera: Esta função detecta o término de utilização do robô e move o robô de forma autônoma para o seu local de origem. Possui como pré-condição: o robô está pronto para receber o sinal de fim de uso. Como pós-condição: O robô está no local de origem.

3. Cenário de utilização do sistema robótico

O robô foi projetado para ser utilizado em ambientes confinados, que possuam um **mapeamento prévio**, não sendo possível utilizá-lo em ambientes externos.

Nesta atividade, iremos utilizar como exemplo de ambiente para utilização do robô uma clínica de repouso para idosos com problemas de locomoção, possuindo quartos com banheiros individuais, um corredor que leva à sala de jantar e uma sala de jantar para as refeições coletivas. Um exemplo para utilização do robô é na **atividade de acompanhamento do paciente ao banheiro.**

Para realização da atividade de acompanhamento do paciente ao banheiro, primeiramente é necessário que o **paciente chame o robô** através de gestos e/ou por voz, o robô, por sua vez, deve despertar, detectar a posição que deve ser alcançada e mover-se para próximo do paciente.

O próximo passo é o **levantamento do paciente**, de modo que o robô detecte o desejo do paciente de se levantar e mova as alças para a altura do paciente. A partir de então, a funcionalidade de gerenciamento de perda de equilíbrio é acionada, para que, c**aso os sensores detectem desequilíbrio**, o robô consiga restabelecer o paciente.

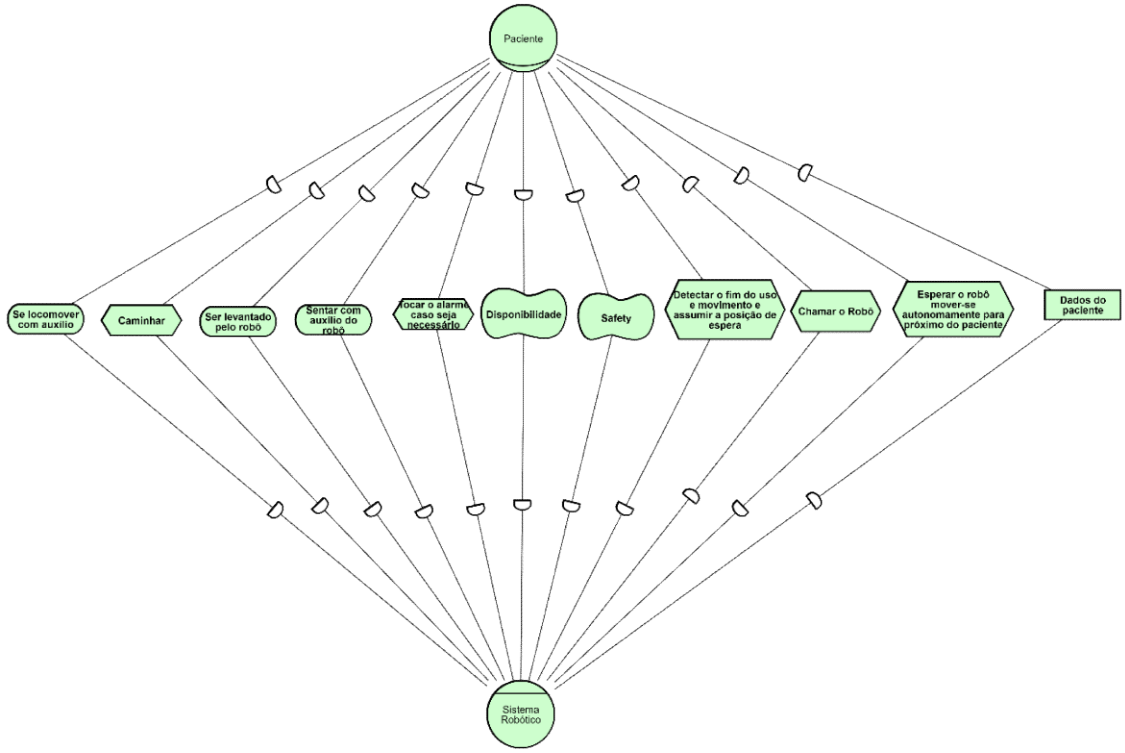
A seguir, o **paciente deve caminhar até o banheiro com auxílio do robô**, que deve estar em modo de caminhada e deve realizar o deslocamento na direção indicada pelo usuário. Durante esta caminhada, o idoso é monitorado de diversas formas: sua postura, perda de equilíbrio, cansaço e até mesmo o tom da voz, além da detecção de fadiga, portanto, caso alguma anormalidade seja detectada a funcionalidade de gerenciamento de alarmes deve ser ativada, para que o alarme seja tocado.

Ao chegar no banheiro, o paciente irá sentar no banheiro, para tanto, o robô deve realizar uma manobra para auxiliar a pessoa a se sentar na posição correta e deverá verificar se possui um assento próximo. Após o término da operação para sentar o paciente, **o robô deverá detectar o fim de uso e se movimentar para a posição de espera**, aguardando o chamado do paciente para auxiliá-lo a levantar e voltar ao quarto.

4. Modelo iStar do sistema robótico MIRAS

4.1 Modelo SD

Figura 62 - Modelo SD Miras Robot



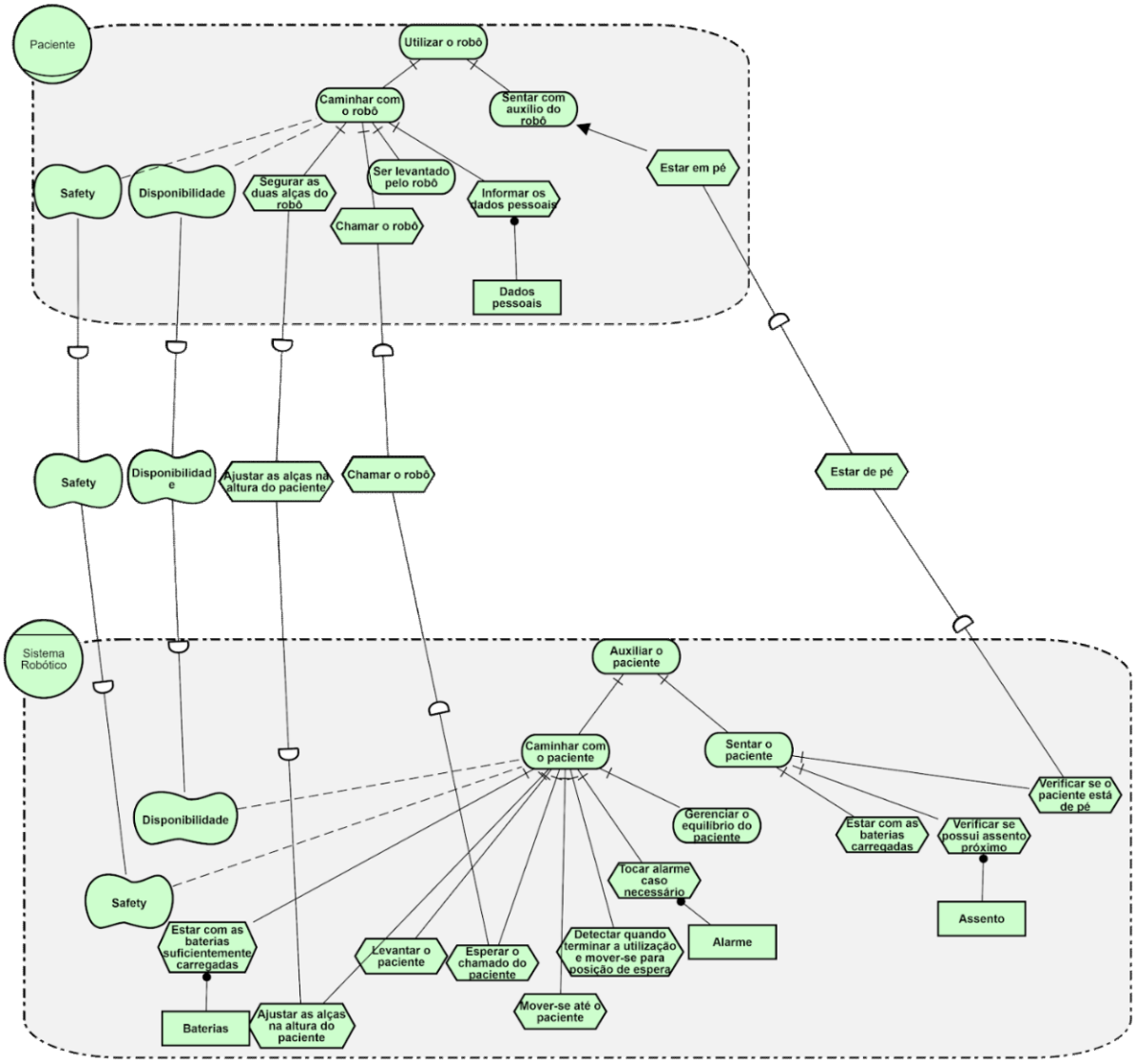
Fonte: Autora (2021)

De acordo com o modelo, podemos observar dois atores: Paciente e Sistema robótico.

O paciente depende do robô para atingir os objetivos de se locomover com auxílio, ser levantado pelo robô, sentar com auxílio do robô. Ademais, o paciente também depende do robô para realização das tarefas de caminhar, tocar o alarme caso alguma anormalidade seja encontrada, detectar o fim do uso e movimento e assumir a posição de espera e esperar o robô mover-se autonomamente para próximo do paciente. O paciente depende também da disponibilidade do robô e que seja *safety*. O robô, por sua vez, depende que o paciente o chame.

4.2 Modelo SR

Figura 63 - Modelo SR MIRAS Robot



Fonte: Autora (2021)

No modelo SR podemos perceber que o paciente tem como objetivo principal utilizar o robô, refinado em dois perigos secundários (1) Caminhar com o robô, para isso é esperado disponibilidade e que seja *safety.* Para atingir este objetivo, o paciente deve segurar as duas alças do robô, chamar o robô, informar os dados pessoais que possui um recurso de dados pessoais. Para atingir o objetivo de caminhar com o robô, espera-se que o paciente cumpra o objetivo de ser levantado pelo robô e (2) Sentar-se com auxílio do robô, para isso, é necessário estar em pé.

O sistema robótico tem como objetivo principal auxiliar o paciente, refinado em dois perigos secundários: (1) Caminhar com o paciente, é esperado que seja realizado de maneira *safety* e que tenha disponibilidade. Ademais, é necessário realizar as seguintes atividades: estar com as baterias suficientemente carregadas, ajustar as alças na altura do paciente, esperar o chamado do paciente, mover-se até o paciente, levantar o paciente, tocar o alarme caso seja necessário, detectar quando termina a utilização para mover-se para a posição de espera e (2) Sentar o paciente, para isso, é necessário a realização de três tarefas: estar com as baterias carregadas, verificar se possui assento próximo e verificar se o paciente está de pé.

A.2 TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO

As informações contidas neste formulário visam firmar acordo por escrito, mediante o qual o aluno autoriza sua participação no experimento Elicit4Safety em Ambiente Acadêmico, com pleno conhecimento da natureza dos procedimentos a que se submeterá para ser participante do estudo e com capacidade de livre arbítrio e sem qualquer coação. Esta participação é voluntária e o participante deste experimento tem a liberdade de retirar seu consentimento a qualquer momento e deixar de participar do estudo, sem qualquer prejuízo ao atendimento a que está sendo ou será submetido.

I – TÍTULO DO TRABALHO EXPERIMENTAL

Elicit4Safety - Utilização da abordagem para auxiliar na modelagem de requisitos iniciais de segurança através do iStar4Safety

II – OBJETIVO DO ESTUDO

Comparação entre a modelagem de requisitos de segurança no iStar4Safety: utilização da abordagem Elicit4Safety para auxílio x modelagem tradicional.

III – INSTITUIÇÃO RESPONSÁVEL

Universidade Federal de Pernambuco (UFPE)

IV – PESQUISADORES RESPONSÁVEIS

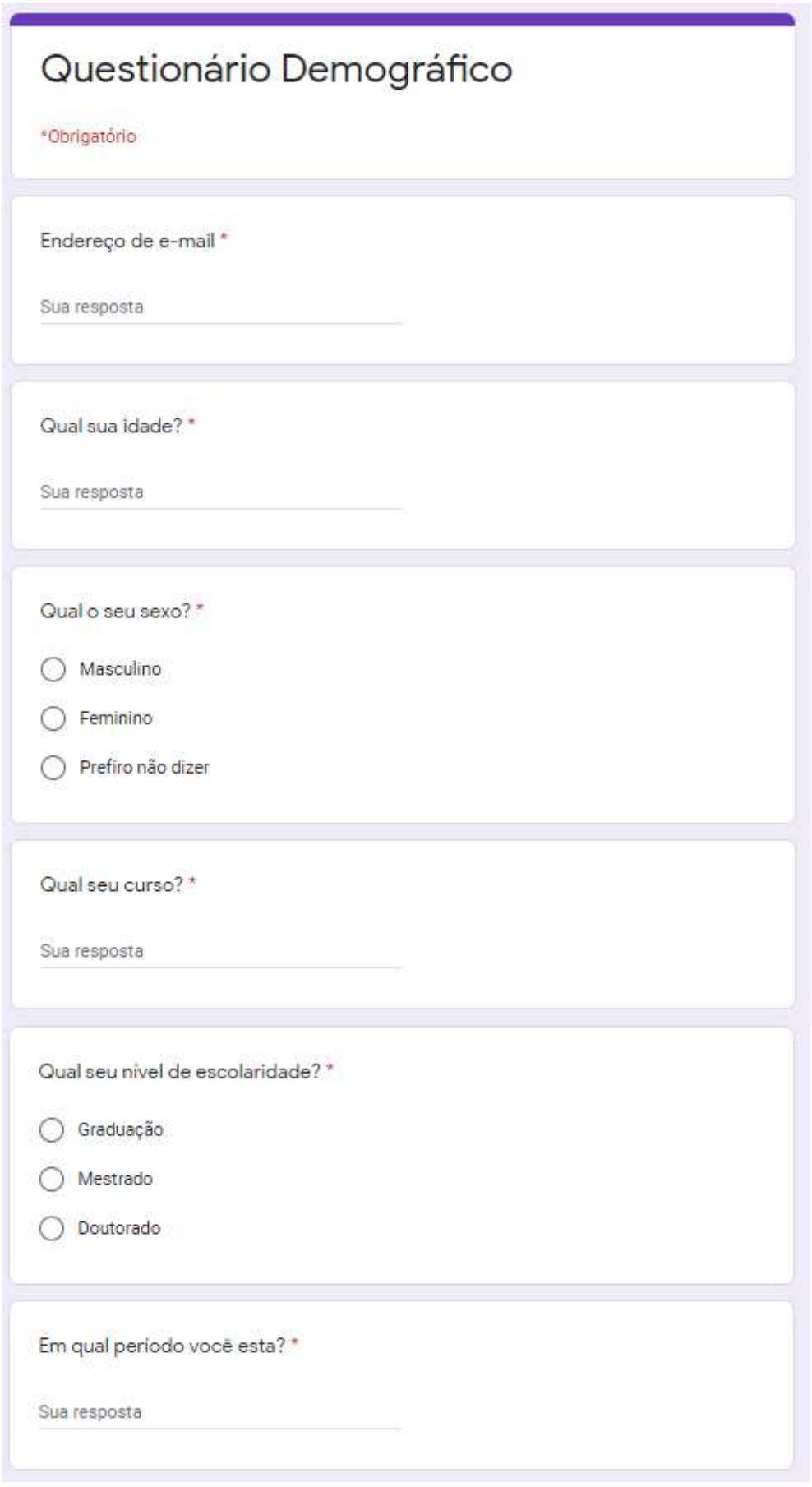
Sthéfanie Dal Magro, Jaelson Castro.

Eu, declaro que, tendo lido as informações acima e suficientemente esclarecido de todos os itens, estou plenamente de acordo com a participação no experimento. Assim, autorizo a execução do trabalho de pesquisa exposto acima e a utilização dos dados gerados por mim para publicação anônima e sumarizada dos resultados.

( ) Concordo

( ) Discordo

A.3 O QUESTIONÁRIO REALIZADO NO APLICATIVO "*GOOGLE FORMS*" COM OS DADOS DEMOGRÁFICOS DOS PARTICIPANTES



Questionário Demográfico

*Obrigatório

Endereço de e-mail *

Sua resposta

Qual sua idade? *

Sua resposta

Qual o seu sexo? *

- Masculino
- Feminino
- Prefiro não dizer

Qual seu curso? *

Sua resposta

Qual seu nivel de escolaridade? *

- Graduação
- Mestrado
- Doutorado

Em qual periodo você esta? *

Sua resposta

Como você definiria seu nível de conhecimento em elicitação de requisitos? *

- ◯ Nenhum
- ◯ Baixo
- ◯ Médio
- ◯ Alto
- ◯ Muito alto

Como você definiria seu nível de conhecimento em Análise Preliminar de Perigos? *

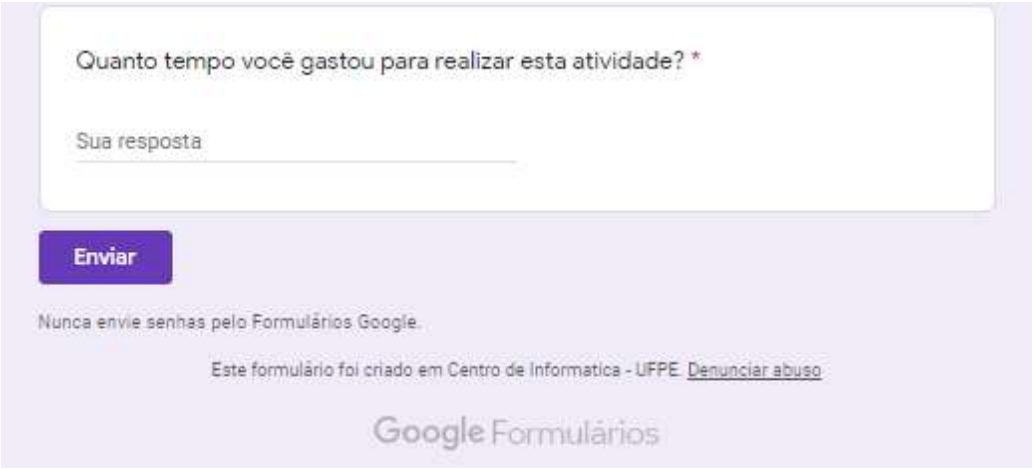
- ◯ Nenhum
- ◯ Baixo
- ◯ Médio
- ◯ Alto
- ◯ Muito alto

Como você definiria seu nível de conhecimento em modelagem iStar? *

- ◯ Nenhum
- ◯ Baixo
- ◯ Médio
- ◯ Alto
- ◯ Muito alto

Como você definiria seu nível de conhecimento em Safety? *

- ◯ Nenhum
- ◯ Baixo
- ◯ Médio
- ◯ Alto
- ◯ Muito alto


Quanto tempo você gastou para realizar esta atividade? *
Sua resposta
Enviar
Nunca envie senhas pelo Formulários Google.
Este formulário foi criado em Centro de Informatica - UFPE. Denunciar abuso
Google Formulários

A.4 QUESTIONÁRIO DE AVALIAÇÃO DA ABORDAGEM

Análise da abordagem Elicit4Safety

Este questionário tem como objetivo avaliar o nível de entendimento e facilidade de uso do sistema Elicit4Safety para auxiliar na descoberta e catalogação de requisitos iniciais de segurança em Sistemas Críticos de Segurança.

As respostas na escala Likert correspondem à:
(1) Discordo totalmente
(2) Discordo parcialmente
(3) Indiferente
(4) Concordo parcialmente
(5) Concordo totalmente

1. Qual o nível de dificuldade em entender a abordagem Elicit4Safety? *

- ◯ Muito fácil
- ◯ Fácil
- ◯ Médio
- ◯ Difícil
- ◯ Muito difícil

2. A utilização da abordagem Elicit4Safety contribuiu para a descoberta de perigos, suas causas e estratégias de mitigação *

	1	2	3	4	5	
Discordo totalmente	◯	◯	◯	◯	◯	Concordo totalmente

3. A utilização da abordagem Elicit4Safety contribuiu para auxiliar a modelagem dos perigos, suas causas e estratégias de mitigação em iStar4Safety *

	1	2	3	4	5	
Discordo totalmente	◯	◯	◯	◯	◯	Concordo totalmente

4. A abordagem Elicit4Safety é um artefato útil nas etapas de elicitação e modelagem de requisitos no contexto dos Sistemas Críticos de Segurança.

	1	2	3	4	5	
Discordo totalmente	○	○	○	○	○	Concordo totalmente

5. Engenheiros de requisitos podem se beneficiar ao utilizar a abordagem Elicit4Safety para descobrir e catalogar os requisitos iniciais de segurança de um sistema crítico *

	1	2	3	4	5	
Discordo totalmente	○	○	○	○	○	Concordo totalmente

6. No geral, estou satisfeito com a facilidade de uso desta abordagem *

	1	2	3	4	5	
Discordo totalmente	○	○	○	○	○	Concordo totalmente

7. Eu me senti confortável utilizando a abordagem Elicit4Safety. *

	1	2	3	4	5	
Discordo totalmente	○	○	○	○	○	Concordo totalmente

8. Foi fácil de aprender a utilizar a abordagem *

	1	2	3	4	5	
Discordo totalmente	○	○	○	○	○	Concordo totalmente

9. As informações fornecidas pela abordagem são fáceis de entender. *

	1	2	3	4	5	
Discordo totalmente	○	○	○	○	○	Concordo totalmente

10. Gostei de utilizar a interface do Elicit4Safety *

	1	2	3	4	5	
Discordo totalmente	○	○	○	○	○	Concordo totalmente

11. No geral, estou satisfeito com a utilização da abordagem Elicit4Safety *

	1	2	3	4	5	
Discordo totalmente	○	○	○	○	○	Concordo totalmente

12. Para qual nivel de profissionais você recomendaria o Elicit4Safety? *

- ○ Para engenheiros de software com experiência em safety
- ○ Para engenheiros de software sem experiência em safety
- ○ Para ambos
- ○ Para nenhum
- ○ Outro: _____

13. Você tem alguma proposta de mudança, melhoria ou comentários relacionados à abordagem Elicit4Safety? *

Sua resposta

Quanto tempo você gastou para realizar esta atividade? *

Sua resposta

Enviar uma cópia das respostas para o meu e-mail.

Voltar | Enviar

Nunca envie senhas pelo Formulários Google.

A.5 UM RESUMO DAS AULAS DADAS VIRTUALMENTE, A FIM DE INSTRUIR OS PARTICIPANTES SOBRE OS TEMAS PERTINENTES AO ASSUNTO.

Terminologias

Definir conceitos comuns é importante para projetos como o desenvolvimento de sistemas críticos que envolve profissionais das mais diversas áreas.

Vilela et al (2017) publicaram um artigo sobre a integração entre a engenharia de requisitos e análise de segurança.

Leveson (1995) afirma que os termos em segurança de sistema não são usados de forma consistente

A partir da utilização destes dois trabalhos, vamos definir os principais termos relacionados à segurança de sistemas.

19

Terminologias

Confiabilidade (Reliability): Probabilidade do componente ou equipamento realizar satisfatoriamente a função definida ao mesmo pelo tempo prescrito e sobre certas condições ambientais
Ex.: O sistema deve enviar um alerta quando a bateria da bomba de insulina estiver com ¼ de sua capacidade -> O sistema envia o alerta.

Falha (Failure): A não-realização ou inabilidade do sistema ou componente realizar a função definida para o mesmo sobre um tempo e sobre condições ambientais específicas. (Evento, comportamento).
Ex.: O sistema deve enviar um alerta quando a bateria da bomba de insulina estiver com 1/4 de sua capacidade -> Ele não envia o alerta na condição prescrita, pelo fato de outro alarme ter sido acionado no momento.

20

Terminologias

Acidente (Accident): Um evento indesejado e não planejado(porém não necessariamente inesperado) que resulta em, no mínimo, um nível específico de perda.
Ex.: O paciente sofre uma overdose de insulina

Incidente (Incident): Um evento que não envolve perdas mas com potencial para tal, sobre diferentes condições.
Ex.: A seringa quebra, porém não afeta a infusão de insulina. Em outras condições poderia levar à underdose de insulina.

21

Terminologias

Perigo (Hazard): Um estado ou conjunto de condições de um sistema, que juntos com outras condições ambientais do sistema, irão levar inevitavelmente a um acidente. O que constitui um perigo depende dos limites definidos para o sistema.
Ex.: Bomba volta inadvertidamente ao estado de fábrica
- O perigo tem duas propriedades importantes: Severidade (ou dano) e probabilidade de ocorrência.
- Nível do perigo = Severidade + Probabilidade

22

Terminologias

Nível de Severidade do Perigo

SEVERITY CATEGORIES		
Description	Severity Category	Mishap Result Criteria
Catastrophic	1	Could result in one or more of the following: death, permanent total disability, irreversible significant environmental impact, or monetary loss equals to or exceeding $10M.
Critical	2	Could result in one or more of the following: permanent partial disability, injuries or occupational illness that may result in hospitalization of at least three personnel, reversible significant environmental impact, or monetary loss equal to or exceeding $1M but less than $10M.
Marginal	3	Could result in one or more of the following: injury or occupational illness resulting in one or more lost work day(s), reversible moderate environmental impact, or monetary loss equal to or exceeding $100K but less than $1M.
Negligible	4	Could result in one or more of the following: injury or occupational illness not resulting in a lost work day, minimal environmental impact, or monetary loss less than $100K.

23

Terminologias

Nível de Probabilidade do Perigo

PROBABILITY LEVELS			
Description	Level	Specific Individual Item	Fleet or Inventory
Frequent	A	Likely to occur often in the life of an item.	Continuously experienced.
Probable	B	Will occur several times in the life of an item.	Will occur frequently.
Occasional	C	Likely to occur sometime in the life of an item.	Will occur several times.
Remote	D	Unlikely, but possible to occur in the life of an item.	Unlikely, but can reasonably be expected to occur.
Improbable	E	So unlikely, it can be assumed occurrence may not be experienced in the life of an item.	Unlikely to occur, but possible.
Eliminated	F	Incapable of occurrence. This level is used when potential hazards are identified and later eliminated.	Incapable of occurrence. This level is used when potential hazards are identified and later eliminated.

24

Terminologias

Risco (Risk): Risco é o nível do perigo combinado com a probabilidade do perigo levar à um acidente (dano) e a exposição, ou duração, ao perigo (latência).

25

Terminologias

Matriz de Avaliação de Risco

RISK ASSESSMENT MATRIX				
SEVERITY / PROBABILITY	Catastrophic (1)	Critical (2)	Marginal (3)	Negligible (4)
Frequent (A)	High	High	Serious	Medium
Probable (B)	High	High	Serious	Medium
Occasional (C)	High	Serious	Medium	Low
Remote (D)	Serious	Medium	Medium	Low
Improbable (E)	Medium	Medium	Medium	Low
Eliminated (F)	Eliminated			

26

Terminologias

- A análise de perigos (Hazard analysis / Safety analysis) envolve somente a identificação de perigos e avaliação do nível do perigo. Já a análise de riscos adiciona a identificação e avaliação de condições ambientais junto com sua exposição ou duração.

- Logo, a análise de perigos é um subconjunto da análise de riscos.

27

Terminologias

Causa de Perigo: É representada por uma condição que sozinha ou associada à outras, é/são suficiente(s) para o perigo relacionado à ela(s) ocorrer. As causas do perigo podem ser controladas ou até eliminadas em alguns casos.
Ex.: No perigo "Bomba volta inadvertidamente ao estado de fábrica" sua causa pode ser devido à uma falha de hardware ou falha de software.

Condição Ambiental: Tratam-se de um conjunto de componentes e suas propriedades, incluindo elementos físicos, culturais, demográficos, econômicos, políticos, regulatórios ou tecnológicos que, apesar de não serem parte do sistema, podem afetar seu comportamento.

28

Terminologias

Requisitos funcionais de segurança: São os requisitos funcionais usados para mitigar ou prevenir os efeitos de falhas identificadas na análise de segurança.

Estratégias de Segurança: São ações que visam mitigar as consequências de um possível acidente. O objetivo dessas ações é eliminar ou reduzir o risco associado a uma situação perigosa. Cada mitigação tem um custo para sua realização, que na maioria das vezes envolve o consumo algum **recurso.**

29

Terminologias

Recurso: Na linguagem iStar recursos são apresentados como entidades informacionais ou físicas que são requeridas pelo ator à fim de realizar uma tarefa. Em Sistemas Críticos de Segurança, recursos são os ativos necessários para o correto funcionamento de requisitos críticos.

30

Acidentes (Accidents)

- Um acidente é um evento não planejado e nem desejado, mas não necessariamente não esperado que resulta em, no mínimo um nível específico de perda.

- Acidentes são geralmente relacionados à requisitos falhos:
  - Mal escritos, incompletos.
  - Ou mesmo características/estados do sistema não elicitadas e não documentadas.

31

Acidentes (Accidents)

- Ao contrário do que muitos pensam, acidentes e grandes catástrofes são mais comumente relacionadas às fases iniciais de desenvolvimento e não na codificação.

- Software correto ou confiável nem sempre é sinônimo de software seguro.

32

Técnicas de Análise de Perigos

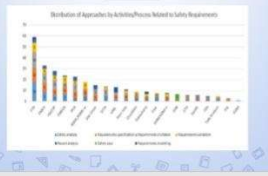
A fim de garantir a segurança do sistema são realizadas análises de perigos, buscando investigar fatores que podem levar à acidentes e como mitigar estes fatores.

Esta etapa permite a categorização da gravidade de acidentes e perigos, bem como a definição da probabilidade em que esses acidentes e perigos podem ocorrer.

Existem diversos técnicas para realização destas análises, dependendo do propósito e momento em que é realizada.

33

Técnicas de Análise de Perigos



34

Fault Tree Analisys (FTA)

O FTA foi desenvolvido em 1961 por H.A. Watson nos Laboratórios Bell e é uma técnica amplamente utilizada em vários contextos. Esta técnica é representada através de uma árvore, com o objetivo de analisar as causas de perigos. O evento do topo da árvore deve ser previsto e identificado primeiramente através de outras técnicas, como exemplo, o HAZOP.

O FTA baseia-se na seleção de um evento principal e na avaliação da combinação de falhas e condições que podem fazer com que o evento principal ocorra.

35

Fault Tree Analisys (FTA)

Os resultados do FTA podem auxiliar os stakeholders a realizar as seguintes atividades:
(1) Verificar a conformidade do projeto com os requisitos de segurança estabelecidos
(2) Identificar as deficiências de segurança do projeto que se desenvolveram apesar dos requisitos existentes
(3) Identificar as falhas de modo comum
(4) Estabelecer medidas preventivas para mitigar ou eliminar deficiências de segurança de projeto
(5) Avaliar a adequação das medidas preventivas estabelecidas e
(6) Estabelecer ou modificar requisitos de segurança adequados para a próxima fase do projeto.

36

Hazard and Operability Analisys (HAZOP)

O HAZOP foi desenvolvido nos anos 60 pela Indústria Química imperial, e é uma técnica baseada em um modelo de teoria de sistemas de acidentes, que assume a causa dos acidentes através de desvios do projeto ou intenções de operação, para tanto, a técnica incentiva o pensamento criativo sobre todas as formas possíveis em que perigos ou problemas operacionais podem surgir.

O HAZOP deve ser executado sistematicamente, considerando cada unidade de processo na planta e cada risco de maneira individual, buscando reduzir a chance de que algo seja esquecido.

37 ★

Hazard and Operability Analisys (HAZOP)

A análise HAZOP envolve a obtenção de uma descrição completa do sistema e o questionamento de cada componente dele, objetivando identificar os desvios de comportamento que possam surgir. A partir da identificação dos desvios é necessário a realização de uma avaliação dos efeitos negativos que esses desvios possam trazer.

A análise HAZOP deve ser realizada com uma equipe multidisciplinar, permitindo que todos os aspectos do sistema sejam observados e questionados.

38 ★

Preliminary Hazard Analisys (PHA)

De acordo com Ericson (2005) é a técnica mais comum para análise de perigos e consiste em uma ferramenta de análise de segurança para identificar perigos, seus fatores, efeitos e níveis de risco e mitigação deles.

O PHA fornece um método eficaz para a descoberta e comparação de perigos do sistema, auxiliando na descoberta dos requisitos iniciais de segurança do sistema.

O PHA é uma técnica rápida e fácil de ser aplicada.

39 ★

Preliminary Hazard Analisys (PHA)

System: Subsystem/Function	①	②	Preliminary Hazard Analysis				Analyst: Date:	③	④
No.	Hazard	Causes	Effects	Mode	MPH	Recommended Action	PMPH	Comments	Status
⑤	⑥	⑦	⑧	⑨	⑩				⑭

Severity: I. Catastrophic; II. Critical; III. Marginal; IV. Negligible

Probability: A. Frequent; B. Probable; C. Occasional; D. Remote; E. Improbable

40 ★

Preliminary Hazard Analisys (PHA)

System: Arc Missile System
Subsystem: Missile Structure Subsystem

Preliminary Hazard Analysis — Analyst: Date:

No.	Hazard	Causes	Effects	Mode	MPH	Recommended Action	PMPH	Comments	Status
PHA-1	Missile structure fails, resulting in unstable missile flight and missile crash	Manufacturing defect; design error	Unstable flight, resulting in crash causing death/injury; missile target	Flight	1D	Use 5:1 safety factor on structure design	1E		Open
PHA-2	Missile body breaks up, resulting in fuel leakage, and subsequent fire, causing fire	Manufacturing defect; design error	Missile fire, causing death/injury	Ground operations	1D	Use 5:1 safety factor on structure design	1E		Open
PHA-3	Missile structure fails during handling, resulting in personnel injury	Manufacturing defect; design error; handling equipment failure	Personnel injury	PHA#7?	2D	Use 5:1 safety factor on structure design Establish SOPs for handling equipment	2E		Open

Page 1 of 4

41 ★

Engenharia de Requisitos e Safety

Engenharia de requisitos é o ramo da engenharia de software que lida com requisitos de software, englobando as atividades de:

- Elicitar, especificar, modelar e validar requisitos (Lapouchnian, 2005)

A elicitação de requisitos para sistemas críticos é um processo bastante oneroso, devido à necessidade de capturar os comportamentos de todos os subsistemas envolvidos e lidar com todas as restrições definidas.

Somado à isso, muitas evidências apontam que erros relacionados à segurança estão mais relacionados à erros em requisitos do que à codificação.

42 ★

Engenharia de Requisitos e Safety

A fase da engenharia de requisitos é a forma mais econômica para corrigir muitos problemas.

A utilização de uma abordagem bem elaborada de engenharia de requisitos no contexto de sistemas críticos de segurança pode trazer alguns benefícios para o desenvolvimento de SCSs:

Atingir as metas de tempo, custo e qualidade

43 ★

Engenharia de Requisitos e Safety

Um requisito de um sistema crítico de segurança é capaz de executar uma ou mais vezes as seguintes funções:

- controlar o funcionamento do hardware crítico de segurança;
- monitorar os estados do sistema com o objetivo de garantir sua segurança;
- detectar riscos e exibir informações relacionadas à proteção do sistema;
- lidar ou responder às prioridades de detecção de falhas;
- computar dados críticos de segurança.

44 ★

Engenharia de Requisitos e Safety

- Tanto as abordagens de engenharia de software, como as da engenharia de segurança devem ser tratadas no processo de engenharia de requisitos, permitindo a participação ativa e comunicação entre os diversos profissionais interessados no desenvolvimento de software seguro e confiável.

- A comunicação satisfatória é um ponto bastante importante no desenvolvimento de sistemas complexos e críticos.

45 ★

Dúvidas??

46 ★

iStar4Safety

1 ★

Revisão dos Conceitos

Acidente: Um evento indesejado e não planejado(porém não necessariamente inesperado) que resulta em, no mínimo, um nível específico de perda.

Perigo (Hazard): Um estado ou conjunto de condições de um sistema, que junto com outras condições ambientais do sistema, irão levar inevitavelmente a um acidente. O que constitui um perigo depende dos limites definidos para o sistema.

Causa de Perigo: É representada por uma condição que sozinha ou associada à outras, é/são suficiente(s) para o perigo relacionado à ela(s) ocorrer. As causas do perigo podem ser controladas ou até eliminadas em alguns casos.

2 ★

Revisão dos Conceitos

Requisitos funcionais de segurança: São os requisitos funcionais usados para mitigar ou prevenir os efeitos de falhas identificadas na análise de segurança.

Estratégias de Segurança: São ações que visam mitigar as consequências de um possível acidente. O objetivo dessas ações é eliminar ou reduzir o risco associado a uma situação perigosa. Cada mitigação tem um custo para sua realização, que na maioria das vezes envolve o consumo algum recurso.

Recurso: Na linguagem iStar recursos são apresentados como entidades informacionais ou físicas que são requeridas pelo ator à fim de realizar uma tarefa. Em Sistemas Críticos de Segurança, recursos são os ativos necessários para o correto funcionamento de requisitos críticos.

3 ★

iStar4Safety

O iStar4Safety é uma extensão do iStar padrão e visa permitir a modelagem dos requisitos iniciais de segurança de um SCSs. Foram adicionados à linguagem padrão do iStar 4 construtores e 1 link capazes de representar as questões voltadas à segurança. São eles:

SafetyGoal (Objetivo de Segurança)

Hazard (Perigo)

SafetyTask (Tarefa de Segurança)

SafetyResource (Recurso de Segurança)

Link Obstructs

4 ★

iStar4Safety

As seguintes características podem ser modeladas através do iStar4Safety:

Acidente: O acidente é a consequência de um perigo (*Hazard*). Logo, subentende-se o acidente quando modelamos o perigo. A modelagem deve ser feita de forma que a negação de um Objetivo de Segurança (*SafetyGoal*) leve à um acidente.

Ex.: Objetivo de segurança: Porta do elevador não abrir quando não permitido.
Acidente: Porta do elevador abre quando não permitido.

5 ★

iStar4Safety

Modelagem de Perigos: É representado pelo construto HAZARD.

O perigo é um obstáculo à realização de um objetivo de segurança.

Ex.: Objetivo de Segurança: Porta do elevador não abrir quando não permitido.
Perigo: Sistema executa de forma errada e abre porta fora do local correto.

Modelagem de Causas de Perigos: É representado pelo construto HAZARD.

As causas de perigos são as causas que levam o perigo à acontecer. São representadas como "Perigos-filhos", ou seja, refinamento de um Perigo Principal.

6 ★

iStar4Safety

Modelagem de Requisitos Funcionais de Segurança: É representado pelos construtos SAFETYTASK e SAFETYRESOURCE.

Requisitos funcionais de segurança irão, em conjunto ou não, formar as Estratégias de Segurança. Através deles, serão representadas as formas de mitigar o perigo.

7 ★

iStar4Safety

Nível de impacto de acidente. É representado através de uma propriedade "nível de impacto" em um Objetivo de Segurança.

O nível de impacto de um acidente deverá ser inserido como propriedade de um Objetivo de segurança.

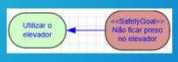
8 ★

Elementos iStar4safety

9 ★

SafetyGoal

SafetyGoal: Um SafetyGoal é um Objetivo Crítico, ou seja, pode contribuir para a ocorrência de acidentes, caso algum perigo associado à ele aconteça.



Cada SafetyGoal pode ser obstruído por de 1 à N Hazards

10 ★

Hazard

Hazard: Um hazard é um obstáculo para que o SafetyGoal se concretize. Ou seja, caso o hazard aconteça, um acidente pode acontecer.

Cada Hazard pode obstruir de 1 à N SafetyGoals

Cada Hazard pode ser refinado por de 0 à N hazards-filhos (causas do perigo)

Os Hazards folha podem possuir de 1 a N Estratégias de Segurança (SafetyTasks e/ou SafetyResources)

11 ★

Link Obstructs

Obstacle (LINK): SafetyGoal e Hazard são ligados através de link OBSTRUCTS. Ou seja, um perigo obstrui que um objetivo de segurança se concretize de forma segura, podendo levar à um acidente.

O Link Obstructs deve ser utilizado para ligar um HAZARD à um SAFETYGOAL

12 ★

Causa de Perigos

Causas de perigos: As causas de perigos são representadas pelo construto Hazard, são consideradas hazards-filhos e podem ser refinadas através da utilização dos links AND ou OR

Cada causa de perigo pode possuir de 1 a N Estratégias de Segurança (SafetyTasks e/ou SafetyResources)

13 ★

Safety Strategies

Estratégias de Segurança: As estratégias de segurança buscam tratar os perigos que possam levar à acidentes.

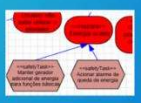
As estratégias de Segurança podem ser representada através dos construtos SafetyTask e SafetyResource.

Todo causa de perigo deve possuir ao menos uma estratégia de segurança.

14 ★

Safety Strategies - SafetyTasks

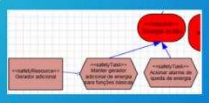
Tarefas de Segurança: Uma SafetyTask representa ações seguras que um ator quer que sejam executadas à fim de mitigar um perigo.



15 ★

Safety Strategies - SafetyResource

Recurso de Segurança: Um SafetyResource deve ser ligado às SafetyTasks através do link Needed-by



16 ★

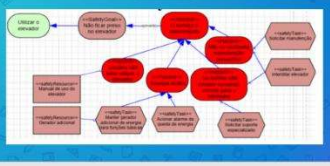
Accident Impact Level

Nível de impacto do acidente: Ele é descrito como propriedade de um SafetyGoal e deve assumir um dos níveis abaixo

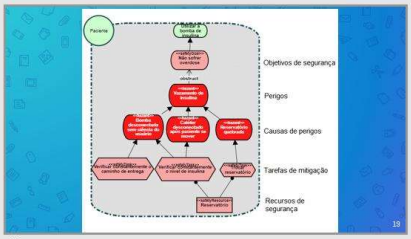
1. Catastrophic (Maior impacto)
2. Hazardous/Severe-Major
3. Major
4. Minor
5. No effect (Sem impacto)

17 ★

Exemplo de mapeamento em iStar4Safety

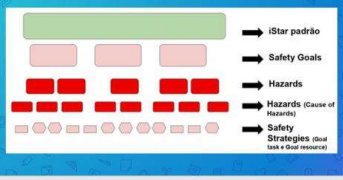


18 ★



19 ★

Visão em camadas – iStar + iStar4Safety



20 ★

Diretrizes

Diretriz 1 – Modelar as funcionalidades relacionadas ao iStar padrão (parte do tipo não-segurança)
Diretriz 2 – Modelar o objetivo de segurança. Um objetivo de segurança é um objetivo crítico que, caso não ocorra como descrito pode provocar um acidente.
Diretriz 3 – Inserir todos os perigos para o objetivo de segurança modelado.
Diretriz 4 – Identificar todas as causas para cada perigo identificado
Diretriz 5 – Definir a estratégia de mitigação para cada perigo-folha (causa).

21 ★

Checklist de Completude

1. Todos os objetivos de segurança tem uma propriedade de nível de impacto do acidente com um valor inserido.
2. Todos os objetivos de segurança tem um ou mais perigos associados.
3. Todos os perigos-raiz estão ligados à um ou mais objetivos de segurança –folha pelo link obstrui.
4. Foram identificadas as causas dos perigos.
5. Todos os perigos-folha estão ligados à uma ou mais estratégias de segurança.

22 ★

Ferramenta piStar4Safety

23 ★

Thanks!

Perguntas?

24 ★

A.6 AVALIAÇÃO DE CONHECIMENTOS ACERCA DE SAFETY E ISTAR4SAFETY

Avaliação de Conhecimentos - Safety

*Obrigatório

Endereço de e-mail *

Seu e-mail

Qual seu nome? *

Sua resposta

Assinale a alternativa que apresenta a correta definição de Perigo (Hazard) *

- ◯ Um evento que não envolve perdas, mas que tem potencial para tal
- ◯ Uma falha de desenvolvimento ou um desvio de um estado desejado ou intencionado
- ◯ Um estado ou conjunto de condições de um sistema que juntos com outras condições ambientais do sistema, irão levar inevitavelmente a um acidente.
- ◯ Um estado indesejado e não planejado que resulta em perda

Assinale a alternativa que apresenta a correta definição de Acidente *

- ◯ Um desvio de um estado desejado ou intencionado que pode causar perdas
- ◯ Um evento que não envolve perdas, mas que tem potencial para tanto
- ◯ Um evento indesejado e não planejado (porém não necessariamente inesperado) que resulta em perda
- ◯ A probabilidade de um componente do sistema falhar

Assinale a alternativa que apresenta a correta definição de Falha *

- ◯ Probabilidade de componente do sistema realizar a funcionalidade do sistema de maneira incorreta
- ◯ A realização de uma função do sistema de maneira correta
- ◯ A não-realização ou inabilidade do sistema ou componente realizar a função definida para o mesmo sobre um tempo e sobre condições ambientais específicas
- ◯ Um desvio de um estado desejado ou intencionado

Assinale a alternativa que apresenta a correta definição de Incidente *

- ◯ Um evento que não envolve perdas mas com potencial para tal, em diferentes condições
- ◯ Um evento indesejado e não planejado que resulta em perda
- ◯ Um evento que envolve grandes perdas
- ◯ Um evento que leva à um acidente

Assinale a alternativa que apresenta a correta definição de Risco *

- ◯ É o nivel do perigo combinado com a probabilidade do perigo levar à um acidente e a expeosição, ou duração, ao perigo.
- ◯ É a probabilidade de um perigo ocorrer
- ◯ É a junção da severidade e probabilidade de que um perigo ocorra
- ◯ É a probabilidade de mitigar um perigo

Assinale a alternativa que apresenta a correta definição de Causa de Perigo *

- ◯ É um conjunto de condições do sistema que leva a um acidente
- ◯ É representado pela condição que sozinha ou associada à outras, é/são suficientes para o perigo relacionado à ela ocorrer.
- ◯ São ações que visam mitigar um perigo
- ◯ Trata-se de um conjunto de condições que visam potencializar um perigo

Assinale a alternativa que apresenta a correta definição de Requisitos Funcionais de Segurança *

- ◯ São os requisitos funcionais que visam aumentar a probabilidade de incidentes
- ◯ São os requisitos funcionais definidos no início do desenvolvimento de um sistema
- ◯ São os requisitos funcionais utilizados para mitigar ou prevenir os efeitos de falhas identificadas em uma análise preliminar de segurança
- ◯ São as restrições de um sistema

Assinale a alternativa que apresenta a correta definição de Recursos *

- ◯ São estratégias utilizadas para mitigar um acidente
- ◯ Na linguagem iStar refere-se à entidades informacionais ou físicas que são requeridas pelo ator à fim de realizar uma tarefa.
- ◯ São componentes que afetam o comportamento do sistema
- ◯ São estratégias utilizadas para mitigar um perigo

Assinale a alternativa que apresenta a correta definição de Estratégias de Segurança *

- ◯ São ações que visam mitigar as consequências de um possível incidente
- ◯ São ações que visam mitigar as consequências de um possível acidente, com o objetivo de eliminar ou reduzir o risco associado a uma situação perigosa.
- ◯ São ações que visam mitigar um perigo
- ◯ São ações que visam mitigar as causas de um perigo

Quais são os construtores do iStar4Safety? *

- ◯ Risco, Condição Ambiental, Link Obstructs, Perigo e Objetivo de Segurança
- ◯ Objetivo de Segurança, Perigo, Tarefa de Segurança, Recurso de Segurança e link Obstructs.
- ◯ Objetivo de Segurança, Risco, Estratégia de Mitigação, Link AND e OR
- ◯ Objetivo de Segurança, Perigo, Tarefa de Segurança, Recursos e Link OR.

Enviar

APÊNDICE B – MAPEAMENTO DAS INFORMAÇÕES DE SEGURANÇA DO MIRAS ROBOT

As informações referentes à segurança do robô MIRAS foram mapeadas utilizando o Elicit4Safety e se obteve o relatório presente nas Figuras 62 à 67.

Relatório Completo

Após preenchimento do questionário, chegou a hora de modelar as informações adquiridas utilizando o iStar4Safety. Acesse aqui! para realizar a modelagem.

A modelagem no iStar4Safety segue uma estrutura de árvore. No primeiro nível da árvore devem ser modelados os objetivos de segurança, o segundo nível trás os perigos, no terceiro nível teremos as causas de perigo, o quarto nível encontram-se as tarefas de mitigação e, por fim, os recursos de segurança.

Wed Mar 24 2021 21:17:03 GMT-0300 (Horário Padrão de Brasília)

Definindo o perfil do Stakeholder:

- **Nome**
- **Em qual empresa você trabalha?**
- **Qual a sua função no desenvolvimento de sistemas críticos de segurança?**
- **Há quanto tempo você trabalha com sistemas críticos de segurança?**

Definindo o perfil da empresa:

- **A empresa em que você trabalha pertence a qual domínio?**
- **A empresa em que você trabalha está estabelecida no mercado há quanto tempo?**

Descobrindo os detalhes do projeto:

- **Sobre o que é o seu projeto?**
- **Qual produto está sendo desenvolvido no seu projeto?**
- **Você possui alguma informação adicional que gostaria de compartilhar?**

Descobrindo os detalhes dos atores do sistema:

- **Qual o nome do Stakeholder ou Sistema?**

  Paciente

  - **Qual objetivo de segurança do Stakeholder / Sistema?**

    Não sofrer acidentes enquanto caminha com o robô

Figura 64 - Relatório (Parte 1)

Fonte: Autora (2021)

- **Qual perigo impede que este objetivo de segurança seja concretizado?**

  Cair enquanto caminha com o robô

  - **Qual seria o efeito (acidente) deste perigo?**

    Queda do paciente

    - **Qual nível de impacto deste efeito (acidente)?**

      muito severo

  - **Se existir, quais as causas deste perigo?**

    Não segurar corretamente nas alças

    - **Quais tarefas de mitigação para esta causa?**

      Segurar firmemente nas alças enquanto caminha com o robô

      - **Quais recursos auxiliam a ação de mitigação?**

        N/A

  - **Quais tarefas de mitigação para este perigo?**

    - **Quais recursos auxiliam a ação de mitigação?**

  - **Se existir, quais as causas deste perigo?**

    Ritmo de caminhada descompassado

    - **Quais tarefas de mitigação para esta causa?**

      Controlar o rítmo de caminhada

      - **Quais recursos auxiliam a ação de mitigação?**

        N/A

- **Qual o nome do Stakeholder ou Sistema?**

  Sistema Robótico

  - **Qual objetivo de segurança do Stakeholder / Sistema?**

    Sentar o paciente

    - **Qual perigo impede que este objetivo de segurança seja concretizado?**

      Não possuir assento próximo

      - **Qual seria o efeito (acidente) deste perigo?**

        Queda do paciente

Figura 65 - Relatório (parte 2)

Fonte: Autora (2021)

- **Qual nível de impacto deste efeito (acidente)?**

  Muito Severo

- **Se existir, quais as causas deste perigo?**

  Falha na detecção de posição do assento

  - **Quais tarefas de mitigação para esta causa?**

    Deve-se verificar se os sensores estão funcionando corretamente

    - **Quais recursos auxiliam a ação de mitigação?**

      Sensores

- **Quais tarefas de mitigação para este perigo?**

  - **Quais recursos auxiliam a ação de mitigação?**

- **Qual objetivo de segurança do Stakeholder / Sistema?**

  Não permitir que o paciente caia

  - **Qual perigo impede que este objetivo de segurança seja concretizado?**

    Alças não estarem na altura correta do paciente

    - **Qual seria o efeito (acidente) deste perigo?**

      Queda do paciente

      - **Qual nível de impacto deste efeito (acidente)?**

        Muito Severo

    - **Se existir, quais as causas deste perigo?**

      Os dados foram informados incorretamente

      - **Quais tarefas de mitigação para esta causa?**

        Os dados devem ser previamente validados

        - **Quais recursos auxiliam a ação de mitigação?**

          Dados corretos

    - **Quais tarefas de mitigação para este perigo?**

      n/a

      - **Quais recursos auxiliam a ação de mitigação?**

    - **Se existir, quais as causas deste perigo?**

Figura 66 - Relatório (parte 3)

Fonte: Autora (2021)

Falha de software

- **Quais tarefas de mitigação para esta causa?**

  O software deve detectar a posição correta do paciente

  - **Quais recursos auxiliam a ação de mitigação?**

- **Qual perigo impede que este objetivo de segurança seja concretizado?**

  Sair da rota estipulada

  - **Qual seria o efeito (acidente) deste perigo?**

    Bater em algum obstáculo

    - **Qual nível de impacto deste efeito (acidente)?**

      Considerável

  - **Se existir, quais as causas deste perigo?**

    Falha de software

    - **Quais tarefas de mitigação para esta causa?**

      Deve ser fornecida a posição em tempo real a um operador humano

      - **Quais recursos auxiliam a ação de mitigação?**

        n/a

  - **Quais tarefas de mitigação para este perigo?**

    N/A

    - **Quais recursos auxiliam a ação de mitigação?**

- **Qual perigo impede que este objetivo de segurança seja concretizado?**

  A bateria não possuir carga suficiente para o percurso

  - **Qual seria o efeito (acidente) deste perigo?**

    O robô parar de funcionar

    - **Qual nível de impacto deste efeito (acidente)?**

      Considerável

  - **Se existir, quais as causas deste perigo?**

    Problemas na especificação do sistema

Figura 67 - Relatório (Parte 4)

Fonte: Autora (2021)

- **Quais tarefas de mitigação para esta causa?**

  O pior caso de consumo de energia deve ser previamente avaliado

  - **Quais recursos auxiliam a ação de mitigação?**

    N/A

- **Quais tarefas de mitigação para este perigo?**

  N/A

  - **Quais recursos auxiliam a ação de mitigação?**

- **Se existir, quais as causas deste perigo?**

  O nível de bateria não é avaliado antes de iniciar a operação

  - **Quais tarefas de mitigação para esta causa?**

    Deve ser realizada uma avaliação prévia da bateria

    - **Quais recursos auxiliam a ação de mitigação?**

      N/A

- **Qual perigo impede que este objetivo de segurança seja concretizado?**

  Colidir com algum objeto / pessoa

  - **Qual seria o efeito (acidente) deste perigo?**

    Machucar o paciente e/ou outra pessoa

    - **Qual nível de impacto deste efeito (acidente)?**

      Muito Severo

  - **Se existir, quais as causas deste perigo?**

    Falha na detecção de obstáculos

    - **Quais tarefas de mitigação para esta causa?**

      Os sensores de colisão devem ser previamente avaliados

      - **Quais recursos auxiliam a ação de mitigação?**

        Sensores de colisão

  - **Quais tarefas de mitigação para este perigo?**

    N/A

    - **Quais recursos auxiliam a ação de mitigação?**

Figura 68 - Relatório (Parte 5)

Fonte: Autora (2021)

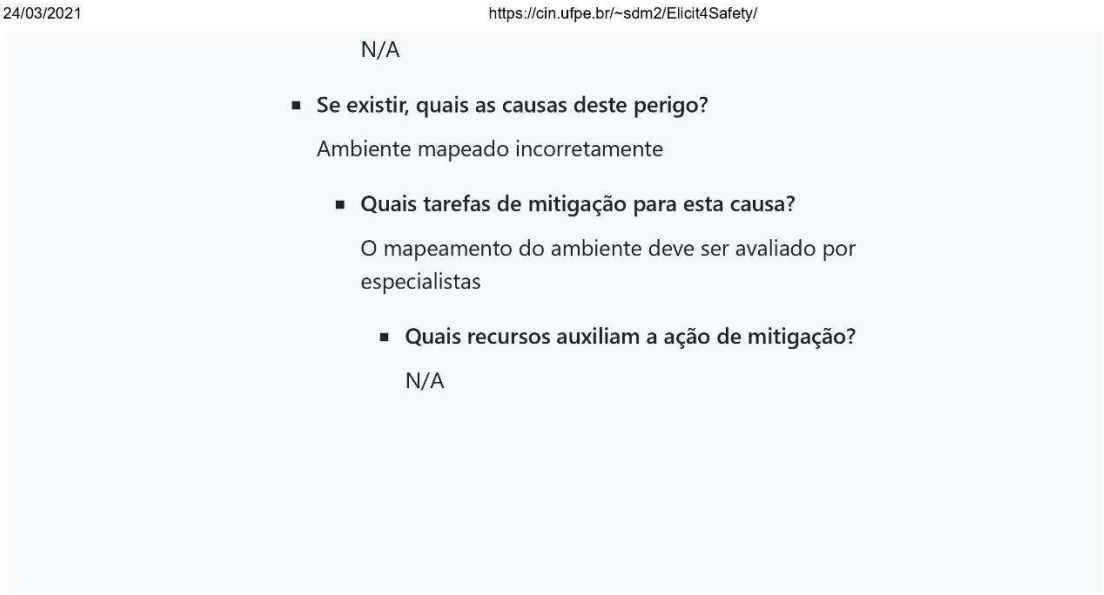


Figura 69 - Relatório (Parte 6)
Fonte: Autora (2021)

A partir do relatório gerado previamente, foi possível modelar as informações de segurança do MIRAS Robot em iStar4Safety, representada na Figura 68.

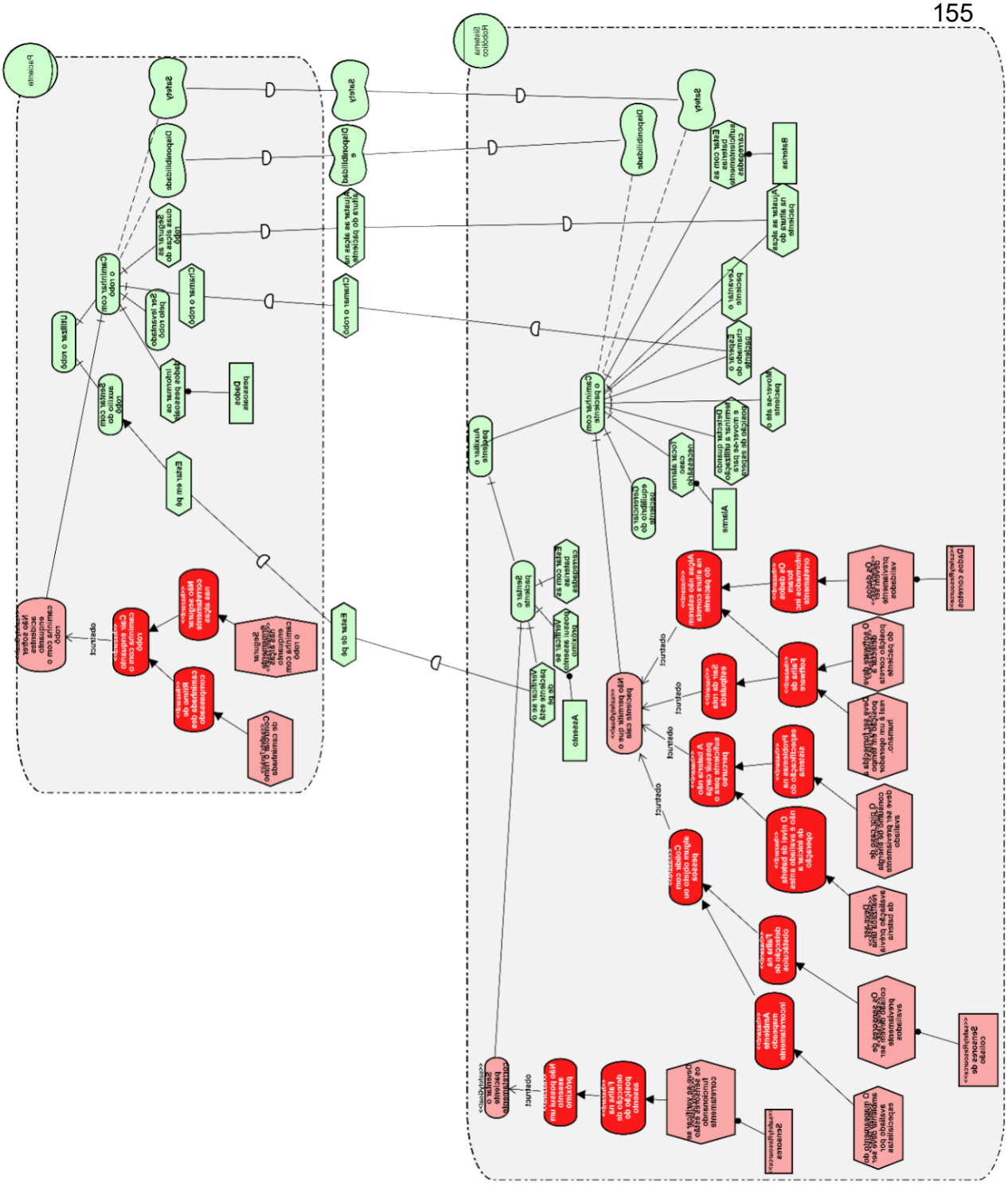


Figura 70 - Mapeamento em iStar4Safety do MIRAS Robot

Fonte: Autora (2021)

APÊNDICE C – DADOS COLETADOS

O Quadro a seguir representa os dados coletados do grupo de controle e experimental.

Participante	Grupo	Elementos Mapeados	Completude	Tempo
#1	Elicit4Safety	19	90	78
#2	Elicit4Safety	9	80	70
#3	iStar4Safety	20	90	52
#4	iStar4Safety	33	90	130
#5	iStar4Safety	37	80	50
#6	iStar4Safety	41	50	75
#7	iStar4Safety	33	60	70
#8	Elicit4Safety	21	80	90
#9	iStar4Safety	23	60	120
#10	Elicit4Safety	37	70	135
#11	Elicit4Safety	24	60	100
#12	iStar4Safety	17	40	80
#13	Elicit4Safety	12	60	45
#14	iStar4Safety	17	60	100
#15	Elicit4Safety	22	100	120
#16	iStar4Safety	31	40	120
#17	Elicit4Safety	14	80	62
#18	iStar4Safety	19	80	70