

# RST-IoT: Um Apoio Computacional para Especificação de Requisitos de Sistemas de Software IoT

Ednilson da S. A. Júnior<sup>1</sup>, Daniella de O. Costa<sup>1</sup>, Rebeca C. Motta<sup>2</sup> e Bruno P. de Souza<sup>2</sup> [0000-0002-1502-7703]

<sup>1</sup> Universidade Federal do Amazonas – ICET/UFAM, Itacoatiara, AM, Brasil  
junioralbuquerque81@gmail.com e daniellacosta@ufam.edu.br

<sup>2</sup> Universidade Federal do Rio de Janeiro – PESC/COPPE, Rio de Janeiro, RJ, Brasil  
{bpsouza, rmotta}@cos.ufrj.br

**Resumo.** A Internet das Coisas (IoT) é resultante da evolução tecnológica constante, e pode ser compreendida como um novo paradigma tecnológico que combina os mundos real e virtual com o objetivo de fornecer sistemas de software em todos os lugares por meio de objetos do cotidiano. Porém, ao mesmo tempo que a tecnologia evolui, maior conhecimento e controle sobre ela devem ser considerados, principalmente nesta área em que o domínio da aplicação não é facilmente compreendido. Pesquisas sobre sistemas IoT e sua composição de dispositivos e aplicações continuam a avançar, todavia, há pouco esforço na evolução da área da Engenharia de Software, especificamente na Engenharia de Requisitos. Esta é uma etapa crítica na construção de sistemas tradicionais; e para sistemas baseados em IoT não é diferente, os quais requerem soluções de requisitos distintos devido a suas novas especificidades. Este trabalho propõe a RST-IoT, uma ferramenta de apoio à especificação de requisitos para sistemas IoT, que tem como objetivo auxiliar engenheiros de software em tal atividade. A RST-IoT foi desenvolvida para apoiar o processo de especificação de requisitos de sistemas IoT, implementada com um design simples visando sua fácil utilização. Foi adotada uma metodologia baseada em experimentação para avaliar a ferramenta. Este trabalho descreve um estudo de viabilidade realizado na indústria e um estudo de observação com estudantes. A RST-IoT se mostrou viável em relação ao tempo de aplicação e à facilidade de uso, bem como obteve evidências iniciais de sua aplicação.

**Keywords:** Engenharia de Requisitos, Especificação de requisitos, Internet das Coisas, Engenharia de Software Experimental.

## 1 Introdução

Na sociedade atual, seu contexto moderno e informatizado é reflexo das grandes revoluções industriais do passado, como a primeira, onde o processo de produção na indústria passou de artesanal para apoiado por máquinas. Seguindo até a terceira, com a informação se tornando uma importante matéria prima. As três revoluções industriais trouxeram a produção em massa, as linhas de montagem, a eletricidade e a tecnologia

da informação, elevando a renda dos trabalhadores e fazendo da competição tecnológica o cerne do desenvolvimento econômico [1].

Como uma nova mudança de paradigma atual, surge à “Quarta Revolução Industrial” ou “Indústria 4.0” [14], apresentada como uma maneira de automatizar os sistemas de software de modo que eles utilizem numerosos recursos de hardware e software. Esses recursos são classificados como “coisas” (ou *things* do inglês) que se conectam por meio de uma rede sem fio, mais precisamente a *internet*, com o intuito de automatizar as tarefas do cotidiano, tanto na vida pessoal, quanto na indústria [14].

Neste contexto, alguns sistemas de software têm surgido como consequência da Indústria 4.0, por exemplo, sistemas de software IoT (Internet das Coisas, ou do inglês *Internet of Things*). Estes sistemas podem ser definidos como o entrelaçamento de objetos conectados, que permitem uma conexão em qualquer lugar e a qualquer momento [2]. Suas principais características são a incorporação de dispositivos (coisas) com capacidades sensoriais e de troca de informações [2]. Dispositivos, sensores, veículos, sistemas e pessoas são exemplos da combinação de coisas, e que podem compartilhar informações entre si. Por meio desse compartilhamento; vida assistida, saúde eletrônica e aprendizado aprimorado são alguns exemplos de possíveis cenários de aplicação nos quais este novo paradigma desempenhará um papel de liderança no futuro próximo [2].

No desenvolvimento de um sistema de software, sua base está em uma especificação de requisitos bem detalhada, sustentada pela Engenharia de Requisitos (ER). De acordo com Ur Rehman *et al.* [3], a ER é parte integrante e fundamental do ciclo de vida de desenvolvimento de software, pois a base para um desenvolvimento bem-sucedido depende da compreensão e abstração de seus requisitos. Para uma melhor compreensão sobre especificação de requisitos, a IEEE [4] define requisito como uma condição ou capacidade necessária por um usuário para resolver um problema e/ou atingir um objetivo, e estes são divididos em duas categorias: funcionais e não-funcionais. Os requisitos funcionais descrevem uma interação entre o sistema e seu ambiente. Enquanto os não-funcionais descrevem uma restrição no sistema que limita a construção da solução para o problema [4].

De acordo com Mahalank *et al.* [5], a especificação de requisitos é considerada a mais importante fase para o desenvolvimento de software de qualidade, porque erros causados por especificações de requisitos inadequadas, podem desacelerar o processo de *design* e implementação do software. Além disso, os erros podem causar danos catastróficos para empresas de construção de software, uma vez que eles podem gerar atraso, retrabalho, e até mesmo, a não entrega do sistema de software [15]. Portanto, esta fase é considerada crítica para o projeto de software.

Tendo em vista que a ER é fundamental na construção do software, e ainda, que sistemas de software IoT possuem características novas [1] [5], especificar requisitos de sistemas de software IoT, talvez, não seja uma tarefa trivial. Portanto, este trabalho tem o propósito de apresentar a RST-IoT (*Requirements Specification Tool for IoT-based Systems*), uma ferramenta para apoiar os engenheiros de software novatos e sem experiência sobre sistemas de software IoT no processo de especificação de requisitos.

Além da Introdução, este artigo está estruturado da seguinte forma. A seção 2 descreve os Trabalhos Relacionados. A seção 3 apresenta a ferramenta RST-IoT. A seção 4 descreve os estudos conduzidos. A seção 5 apresenta alguns das ameaças a

validade dos estudos bem como as limitações da pesquisa. Por fim, a seção 6 apresenta a conclusão e os trabalhos futuros.

## 2 Trabalhos Relacionados

Alguns trabalhos relacionados a ER específicos para sistemas de software IoT têm sido apresentados na literatura técnica. Isso mostra que a área tem proposto abordagens, métodos e técnicas para apoiar a engenharia de requisitos destes novos tipos de sistemas de software. Martins *et al.* [6] propôs ‘TERASE: *Template* para Especificação de Requisitos de Ambiente em Sistemas Embarcados’, cujo objetivo apresentar um modelo para especificação de requisitos de ambiente em sistemas embarcados. Neste modelo, engenheiros de sistemas podem melhorar a especificação de requisitos ambientais, bem como o processo de comunicação entre hardware e software. Os sistemas que são tidos como foco para o *template* TERASE são aqueles construídos com processamento que tem como base microcontroladores, sendo esses sistemas embarcados voltados à linha de eletrodomésticos e produtos que utilizam um ou mais microcontroladores na execução do software. O TERASE foi avaliado por meio de um estudo experimental, utilizando um sistema de *datalogger*, que coleta e armazena leituras de temperatura de um ambiente. Como resultado deste trabalho, os estudos apresentaram benefícios aos engenheiros de sistemas adotarem um *template* para especificar requisitos para sistemas embarcados.

No trabalho de Costa *et al.* [7], ‘*Specifying Functional Requirements and QoS Parameters for IoT Systems*’, é apresentado a IoT-RML, uma linguagem de modelagem que visa auxiliar engenheiros de software na especificação de requisitos para sistemas de software IoT. O principal objetivo é investigar como os requisitos de várias partes interessadas podem ser especificados com precisão utilizando um modelo de requisitos, sendo considerado os conflitos e influências que podem existir entre os requisitos. Com isso, Costa *et al.* [7] apresentam meios para especificar, além de apresentarem a linguagem como um meio de análise de conformidade dos requisitos por meio de simulação. O modelo IoT-RML é proposto com base no modelo de domínio de requisitos IoT, e concebido como um perfil SysML, baseado na *Unified Modeling Language* (UML), que permite especificar graficamente sistemas complexos, incluindo dispositivos de hardware e componentes de software.

O trabalho de Reggio [8], ‘*A UML-based Proposal for IoT System Requirements Specification*’, tem como objetivo apresentar uma versão preliminar de um método (IoTReq) para obter e especificar requisitos de um sistema de software IoT. O método é dividido em tarefas, onde a primeira é a modelagem do domínio da aplicação utilizando a UML e seguindo o paradigma orientado a serviços. A partir dessa tarefa, os objetivos do sistema de software IoT a serem construídos são elicitados e especificados, utilizando novamente a UML e estendendo o modelo de domínio, produzindo uma especificação dos requisitos funcionais.

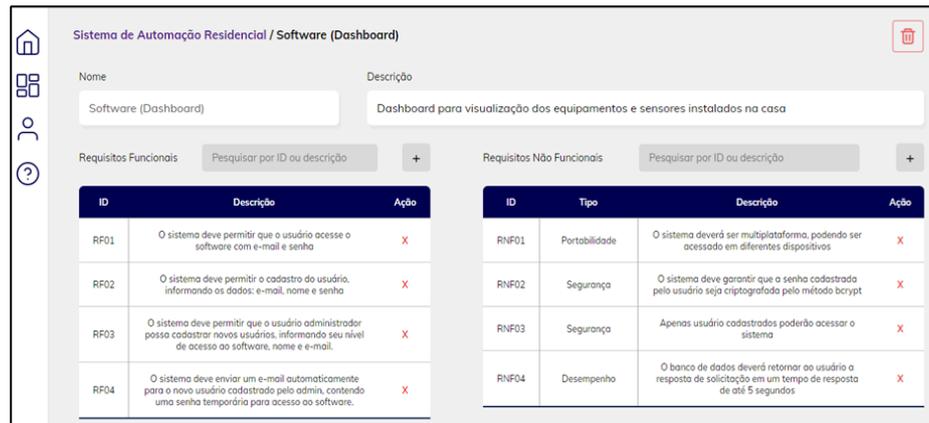
Da Silva *et al.* [9], em ‘*Uma Tecnologia para Apoiar a Engenharia de Requisitos de Sistemas de Software IoT*’, apresentam uma tecnologia, denominada de RE<sub>IoT</sub> (*Requirements Engineering for software systems based on IoT*), com o objetivo de

apoiar a Engenharia de Requisitos no contexto de sistemas de software IoT. A base tecnológica da RE<sub>IoT</sub> é composta por duas técnicas, o SCENAR<sub>IoT</sub> e SCENAR<sub>IoT</sub>CHECK, utilizadas para especificar e inspecionar requisitos de sistemas de software IoT. Com isso, a RE<sub>IoT</sub> fornece *templates* que apoiam as fases de elicitação, análise, especificação e verificação. A RE<sub>IoT</sub> tem sido avaliada experimentalmente em projetos de software no âmbito real.

### 3 RST-IoT: A Requirements Specification Tool for IoT-based Systems

Esta seção apresenta a ferramenta RST-IoT, com suas características e o processo de seu desenvolvimento. O desenvolvimento foi realizado utilizando como tecnologias principais: *Figma*, *NodeJs*, *Javascript* e *ReactJS*.

Esta ferramenta considera a cooperação de todos os integrantes da equipe de desenvolvimento para criação dos requisitos tanto funcionais quanto não-funcionais. Então, todos os membros da equipe devem se cadastrar com ‘usuário’ e ‘senha’ e serem associados a um projeto de software, dentro da ferramenta. A Figura 1 apresenta a tela dos Requisitos Funcionais e Não-funcionais da RST-IoT.



**Figura 1.** Tela dos Requisitos Funcionais e Não-Funcionais.

Para abordar a especificação de requisitos na RST-IoT, deve seguir a organização em projetos, que devem ser cadastrados com nome, tipo, domínio e sua descrição. A Figura 2 apresenta um exemplo da tela da RST-IoT, na qual é possível cadastrar um projeto.

**RST-IoT** Sair

**Criar Projeto**

Nome: Sistema de Automação Residencial

Tipo do sistema: IoT

Domínio do Sistema: Lazer

Descrição: O sistema a ser desenvolvido deve trazer praticidade, conforto e segurança na execução das tarefas com baixo custo de investimento, e ter as seguintes funcionalidades: interconectar todo o sistema de iluminação de uma residência para que possa ser controlado.

**Cadastrar Projeto**

**Figura 2.** Tela de cadastro de projeto.

Após o projeto ser cadastrado, é possível cadastrar seus subsistemas (ou módulos), com o objetivo de seguir uma visão modular, separando em software, hardware e outros módulos que devam compor o projeto. Esta organização visa facilitar a descrição e organização dos requisitos, uma vez que sistemas de software IoT possuem características de modularização. A Figura 3 mostra uma tela com a divisão de módulos, associação dos membros da equipe ao projeto e o objetivo do sistema.

**Sistema de Automação Residencial** Download PDF

Módulos: Software (Dashboard), BrokerMQTT, Módulo Iluminação, Módulo Alarme

Membros:

Nome	Email
Membro 1	membro1@rstiot.com
Membro 2	membro2@rstiot.com
Membro 3	membro3@rstiot.com

Domínio: Lazer

Tipo: IoT

Descrição: O sistema a ser desenvolvido deve trazer praticidade, conforto e segurança na execução das tarefas com baixo custo de investimento, e ter as seguintes funcionalidades: interconectar todo o sistema de iluminação de uma residência para que possa ser controlado.

**Figura 3.** Visualização das informações do projeto.

A RST-IoT está disponível no seguinte endereço eletrônico: <https://bit.ly/39V0bHw>. O usuário e senha para acesso estão disponíveis em um arquivo nomeado de “README”. Um tutorial também foi criado para mostrar como acessar a RST-IoT e está na pasta nomeada “Tutorial da RST-IoT”.

A RST-IoT vem com um esforço inicial de incorporar ferramentas de apoio a automatização de requisitos específicos para sistemas de software IoT. Ela traz um ambiente inovador e colaborador para as empresas e equipes de desenvolvimento.

## 4 Avaliando a RST-IoT por meio de Estudos Experimentais

Esta seção descreve dois estudos experimentais realizados na avaliação da RST-IoT. O primeiro estudo foi realizado para verificar a viabilidade da ferramenta em um projeto real de construção de sistemas de software IoT. Os resultados mostraram que a RST-IoT necessitava de correções, e, conseqüentemente, foi evoluída para uma segunda versão. A segunda versão foi avaliada com um estudo observacional, tendo o objetivo de observar como o RST-IoT foi aplicada por estudantes de graduação e sem experiência com sistemas de software IoT.

### 4.1 Estudo de Viabilidade

#### 4.1.1 Planejamento do Estudo

Primeiramente, faz-se necessário verificar se a RST-IoT é viável. Com esta finalidade, o planejamento do estudo de viabilidade foi realizado visando avaliar a aplicabilidade da ferramenta RST-IoT no processo de especificação de requisitos para sistemas de software IoT. Assim, este estudo foi estruturado utilizando como base o paradigma GQM (*Goal Question Metric*) proposto por Basili e Rombach (1988).

Nesta etapa da definição do estudo, a questão geral foi definida para que a ferramenta fosse avaliada por meio da coleta e análise dos dados qualitativos. Sendo assim, a questão principal: “*A utilização de uma ferramenta auxilia os engenheiros de software novatos no processo de especificação dos requisitos para sistemas IoT?*” A Tabela 1 apresenta a estruturação do estudo.

**Tabela 1.** Objetivo do Estudo de Viabilidade

<b>Analisar</b>	RST-IoT
<b>Com o propósito de</b>	Caracterizar
<b>Com respeito a</b>	Apoiar o processo de especificação de requisitos
<b>Do ponto de vista</b>	Do pesquisador
<b>No contexto de</b>	Engenheiros de Software novatos realizando especificação de requisitos para sistemas de software IoT

**Estudo Piloto:** antes de iniciar o estudo principal, um estudo piloto foi realizado para verificar a qualidade do material que, posteriormente, seria dado aos participantes. Os materiais foram: termo de consentimento de participação do estudo, caracterização dos participantes, questionário pós utilização da ferramenta. No estudo piloto, três alunos de graduação do curso de Engenharia de Software foram convidados para participar. Após a execução do estudo piloto, eventuais melhorias no material foram realizadas.

**Participantes:** Para execução deste estudo experimental, foram convidados os membros do Projeto PROCAP, um projeto de pesquisa e desenvolvimento na área da Indústria 4.0, no qual inclui alunos de graduação do Instituto de Ciências Exatas e Tecnologia (ICET) e engenheiros de software de um instituto de tecnologia em Manaus – Amazonas, Brasil. Participaram deste estudo quatro engenheiros de software que

estavam alocados para este projeto. A Tabela 2 mostra a caracterização dos participantes individualmente.

**Escopo do Projeto:** O projeto PROCAP em parceria do ICET, teve como objetivo automatizar seu processo de produção. PROCAP é uma empresa do ramo de eletrônicos e informática de Manaus e se trata de um projeto de pesquisa e desenvolvimento utilizando tecnologias de visão computacional, sistemas embarcados e internet das coisas.

**Tabela 2.** Caracterização dos participantes do estudo de viabilidade

Nº	Experiência em desenvolvimento de Software	Conhecimento sobre desenvolvimento de software IoT, sistemas embarcados, etc.	Experiência com Engenharia de Requisitos
1	média	baixa	média
2	alta	alta	alta
3	média	média	baixa
4	média	média	baixa

**Materiais Utilizados:** foram disponibilizados o Formulário de Caracterização dos Participantes; o Termo de Consentimento e o Questionário pós utilização da RST-IoT. Todos os materiais foram compartilhados online através do *Google Forms*. Os Formulários utilizados nessa pesquisa estão disponíveis <https://bit.ly/39V0bHw>. Os Formulários estão disponibilizados na pasta nomeada “documentos utilizados”.

#### 4.1.2 Execução do Estudo de Viabilidade

**Execução do Estudo:** Primeiramente, foi realizado um treinamento detalhando conceitos de ER e sistemas IoT. O treinamento foi dado por meio de aulas virtuais intituladas “*Práticas em Engenharia de Software em Sistemas de Software IoT*”, entre os dias 15 e 22 de agosto de 2020, de forma online via *Google Meet*; devido a pandemia de COVID-19. O objetivo desse treinamento foi mostrar boas práticas de construção de software, bem como introduzir as características de sistemas de software IoT.

Cada participante preencheu o formulário de caracterização. Após as aulas, as atividades do projeto foram explicadas e alocadas a equipe de desenvolvimento. Cada membro da equipe recebeu ‘usuário’ e ‘senha’ do projeto do sistema baseado em IoT, e utilizaram a ferramenta RST-IoT para especificar requisitos funcionais e não-funcionais.

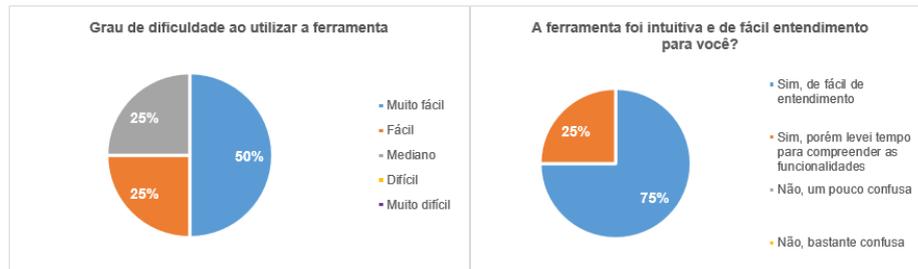
O formulário continha nove questões relacionadas à utilidade e facilidade de uso da RST-IoT. Para a realização da análise qualitativa, foram verificados os formulários preenchidos pelos participantes que utilizaram a ferramenta.

#### 4.1.3 Resultados do Estudo de Viabilidade

Os resultados do estudo de viabilidade foram de caráter qualitativo. Com a análise das respostas dos formulários, pode-se determinar que o grau de dificuldade em utilizar a RST-IoT (Figura 4): dois dos participantes afirmaram que foi “muito fácil” utilizar a

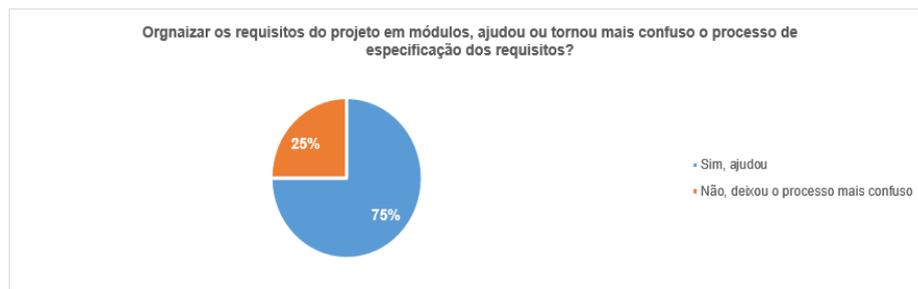
ferramenta RST-IoT, enquanto um participante afirmou que foi “fácil” e outro afirmou que RST-IoT foi “mediana”.

Durante a utilização da ferramenta, três dos participantes afirmaram que a ferramenta foi de fácil entendimento, já um participante relatou que demorou um pouco mais de tempo para compreender as funcionalidades dela. A Figura 4 mostra a percepção dos participantes.



**Figura 4.** Percepção dos participantes em relação à dificuldade da utilização da ferramenta.

Com relação ao auxílio que a ferramenta organiza os requisitos em módulos, três participantes consideraram que a RST-IoT apoiou no processo, e um participante mencionou que ela deixou o processo um pouco confuso, conforme mostra a Figura 5.



**Figura 5.** Organização dos requisitos na ferramenta.

## 4.2 Estudo de Observação

O estudo de observação foi realizado com o objetivo de compreender a utilização da ferramenta no processo de especificação na percepção dos usuários, e responder à pergunta “*As etapas do processo de especificação de requisitos na RST-IoT fazem sentido?*”. Os resultados se mostraram importantes para compreender a utilização da ferramenta no processo de especificação de requisitos por estudantes de engenharia de software. Além disso, este estudo foi útil para identificação de recomendações e correções, objetivando melhorar a qualidade e facilitar a utilização da ferramenta.

### 4.2.1 Planejamento do Estudo de Observação

Um segundo estudo experimental foi realizado visando dados mais sólidos e com um aprimoramento da RST-IoT após *feedback* do primeiro estudo. O planejamento deste

estudo foi realizado para avaliar a experiência dos usuários ao utilizarem a segunda versão da RST-IoT no processo de especificação de requisitos para sistemas de software IoT. Este estudo foi estruturado com base no paradigma GQM (1988) de acordo com a Tabela 3.

**Tabela 3.** Estruturação do Estudo de Observação.

<b>Analisar</b>	A Segunda Versão da RST-IoT
<b>Com o propósito de</b>	Compreender
<b>Com respeito a</b>	Utilização no processo de especificação de requisitos IoT
<b>Do ponto de vista</b>	Do pesquisador
<b>No contexto de</b>	Estudantes de graduação em engenharia de software realizando especificação de requisitos de sistemas de software IoT

**Estudo Piloto:** Foi realizado um estudo piloto para este estudo, tendo o mesmo propósito que do estudo de viabilidade (seção 4.1), onde foi verificar a qualidade dos materiais propostos para o estudo de observação.

**Participantes:** Para a execução do estudo de observação, foram abertas inscrições para um minicurso sobre “*Práticas de Engenharia de Software no Contexto IoT*”. As vagas para o minicurso eram voltadas para estudantes da área da computação que já estivessem cursados a disciplina de Engenharia de Software. Com isso foi obtido um total de 30 inscrições, dos quais 14 assinaram o termo de consentimento para participação da pesquisa, e apenas sete participaram do estudo até o final. A Tabela 4 apresenta a caracterização dos participantes deste estudo.

**Tabela 4.** Caracterização dos participantes do segundo estudo.

Nº	Experiência em desenvolvimento de Software	Conhecimento sobre desenvolvimento de software IoT	Experiência com Especificação de Requisitos
1	média	baixa	baixa
2	média	baixa	média
3	média	baixa	baixa
4	média	média	baixa
5	baixa	nenhuma	nenhuma
6	média	baixa	baixa
7	alta	baixa	alta

**Recursos Utilizados:** foi realizado um minicurso entre dias 16 e 19 de outubro de 2020, de forma online via *Google Meets*; foi elaborado o formulário de caracterização dos participantes; o Termo de consentimento; e o questionário pós-aplicação do estudo experimental com os materiais disponibilizados de modo similar ao primeiro estudo. Os Formulários utilizados nessa pesquisa estão disponíveis <https://bit.ly/39V0bHw>. Os Formulários estão disponibilizados na pasta nomeada “documentos utilizados”.

**Escopo do minicurso:** O minicurso teve como objetivo apresentar os principais conceitos da Engenharia de Software voltado para o contexto IoT, visando preparar os

participantes para uma atividade relacionada a especificação de requisitos de um sistema baseado em IoT.

#### 4.2.2 Execução do Estudo de Observação

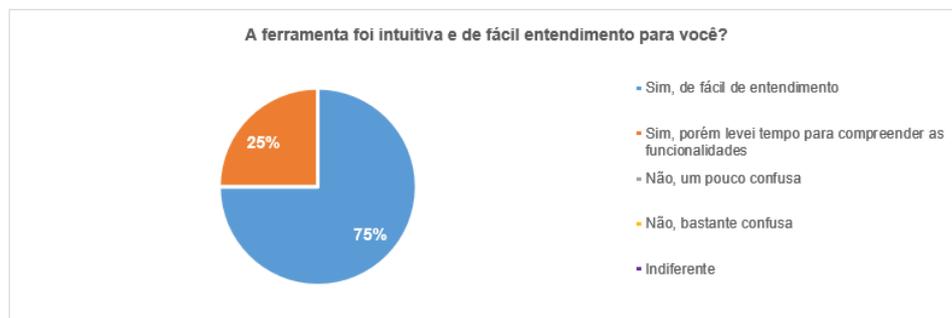
Inicialmente, os participantes responderam o formulário de caracterização, que continha questões para verificar o conhecimento em desenvolvimento de software, desenvolvimento de software IoT e a experiência com especificação de requisitos. Com esses dados foi possível relacioná-los ao grau de dificuldade em utilizar a RST-IoT, conforme mostra a Tabela 4.

A utilização da ferramenta foi realizada individualmente por cada participante, onde eles receberam seu acesso com usuário e senha, e foi apresentado um problema real de um sistema de software IoT para especificarem dos requisitos. O pesquisador autor da RST-IoT participou realizando um treinamento prévio da mesma, além de apresentar o problema proposto. Quando os participantes tinham dúvidas, eles eram orientados utiliza a plataforma *Google Classroom* e e-mail, e assim proceder da maneira que acreditavam ser a mais correta. A maior parte das dúvidas foi relacionado ao acesso a RST-IoT.

#### 4.2.3 Resultados do Estudo de Observação

Após cada participante terminar sua atividade de especificação de requisitos, era solicitado ao mesmo que respondesse um questionário, com o objetivo de avaliar a utilização da ferramenta. Algumas questões foram levantadas com o objetivo de verificar a relevância da ferramenta, seu auxílio no processo de especificação de requisitos, o entendimento e o impacto em organizar os requisitos por módulos.

De acordo com os resultados apresentados na Figura 6, para 62,5% dos participantes a RST-IoT se mostrou fácil de utilizar, porém levaram um pouco de tempo compreender as funcionalidades da mesma, enquanto para 25% foi de fácil entendimento, e somente 12,5% afirmaram que a RST-IoT é um pouco confusa. Com isso, observa-se que a RST-IoT necessita de uma atenção maior quanto a facilitar a compressão das suas funcionalidades para os usuários. Uma possível melhoria seria incluir um tutorial com acesso guiado ao primeiro acesso do usuário na ferramenta.



**Figura 6.** Resultados sobre facilidade de entendimento da ferramenta.

Uma estratégia adotada para organizar os requisitos de um projeto IoT na RST-IoT, foi a utilização de módulos para agrupar requisitos de um mesmo domínio, e na Figura 7, pode-se observar que a percepção dos participantes quanto a esta estratégia auxiliou no processo de especificação de requisitos, onde 62,5% afirmam que esta organização por módulos ajudou no processo, 25% afirmam que deixou o processo um pouco confuso e 12,5% ficaram indiferentes quanto a questão.



**Figura 7.** Resultados sobre a organização dos requisitos em módulos.

Além dos resultados do formulário, realizamos algumas perguntas abertas, a fim de obter um melhor entendimento sobre o uso da ferramenta. Assim, foi possível coletar comentários pertinentes a ferramenta, no qual obtiveram-se as seguintes respostas:

*“A ferramenta é válida e simples, porém confusa. Acredito que seja necessário detalhar mais cada função da ferramenta para melhor entendimento. Um guia de utilização seria ideal também.”* (Participante 8).

*“Poderia existir um campo para editar os dados preenchidos na Descrição do módulo e dos Requisitos. Gostei muito da ferramenta, passarei a usar e recomendar.”* (Participante 8).

*“A ferramenta pareceu bem promissora, acredito que com um pouco mais de tempo dedicado ao aprendizado da mesma, o processo de elicitação de requisitos seria bem mais facilitado. Acredito que uma documentação de como fazer melhor uso das funcionalidades da ferramenta também seria de grande ajuda.”* (Participante 5).

*“Após a criação do projeto fiquei um pouco confusa com aquele “Módulo”, você poderia colocar exemplos de módulos para a pessoa ter uma ideia do que cadastrar.”* (Participante 4).

Com base nessas respostas, a Tabela 5 apresenta propostas de melhorias para a ferramenta, com a coleta do entendimento na utilização da RST-IoT para aprimoramento. Além disso, foi incorporado um conjunto de novas funcionalidades com o objetivo de melhorar a RST-IoT.

**Tabela 5.** Extrato de algumas melhorias extraídas do Estudo de Observação.

<i>Feedback</i>	<i>Melhoria Proposta</i>
Detalhar mais cada função e criação de um guia de utilização.	Acrescentar à ferramenta após o primeiro login, um tour guiado passando por todas as funcionalidades da RST-IoT.
Edição dos dados preenchidos na descrição do módulo e requisitos.	Acrescentar à ferramenta campos para a edição dos dados inseridos pelo usuário, conforme um nível de permissão gerenciado pelo dono do projeto.
Exemplos de módulos para cadastrar.	Acrescentar à ferramenta mais exemplos de módulos ou permitir a criação dos módulos em um segundo momento, podendo relacionar os requisitos cadastrados aos módulos numa segunda etapa, uma vez que sistemas IoT são bastante complexos para se definirem todos os componentes e “módulos” existentes.

## 5 Ameaças a Validade e Limitações da Pesquisa

Os estudos experimentais possuem ameaças que podem influenciar na validade de seus resultados, sendo a validade de um experimento diretamente associada ao nível de confiança que se pode ter em um processo de investigação experimental como um todo [10]. A **Validade Interna**, foi considerada em relação ao treinamento, uma vez que não houve garantia que todos os participantes adquiriram o conhecimento necessário para a execução das atividades. É importante destacar que este estudo foi realizado de maneira assíncrona, não podendo ser controlada as variáveis de contexto, enquanto os participantes realizavam a especificação de requisitos na RST-IoT remotamente.

Na **Validade Externa**, o estudo envolveu um ambiente acadêmico (Estudo de Observação) e utilizou um projeto real (Estudo de Viabilidade) para especificação de requisitos. Não é possível generalizar os resultados para outros contextos. Mas, os resultados do estudo podem servir como indicativos de viabilidade ao utilizar a RST-IoT em apoiar o processo de especificação de requisitos para sistemas IoT. Por mais que no Estudo de Observação, os participantes tenham sido estudantes de graduação, é considerado que eles possuam o mesmo grau de conhecimento que engenheiros de software novatos [11].

Na **Validade de Conclusão**, foi considerado o tamanho reduzido da amostra como um risco, pois a amostra foi pequena em ambos os estudos, limitada e homogênea, dificultando a conclusão dos resultados. Dessa forma, os resultados não podem ser considerados conclusivos, mas como indicativos de evidências.

Este estudo apresentou algumas limitações no que se refere à amostra e população, pois contou com dois estudos com amostras reduzidas, devido aos estudos serem realizados de maneira assíncrona, devido ao momento de pandemia do Covid-19. Outra limitação foi o fato de a população presente em um estudo ser composta por estudantes de graduação com pouca experiência em especificação de requisitos de sistemas de software IoT, tendo em vista ser um tema pouco abordado ainda na comunidade.

## 6 Considerações Finais e Trabalhos Futuros

Este trabalho apresentou uma pesquisa relacionada a especificação de requisitos para sistemas de software IoT. Assim como acontece com outros domínios de aplicações, o objetivo final de um sistema IoT é atender aos requisitos de seus vários interessados. No entanto, a especificação de requisitos para sistemas IoT não é uma tarefa trivial, devido as características novas desses sistemas [7].

Pode ser visto diversas pesquisas voltadas para elicitación de requisitos [7], descrição de cenários IoT [13], verificação de requisitos IoT [12], no entanto, na literatura não se tem visto pesquisas voltadas para criação de um apoio computacional para especificação de requisitos de sistemas de software IoT. Então, esta pesquisa veio com um esforço inicial de inserir um apoio automatizado (RST-IoT) para engenheiros de software a especificarem requisitos de sistemas IoT. Através da RST-IoT, pode-se notar que os engenheiros de software especificaram requisitos de sistemas de software IoT com facilidade, além do uso intuitivo da ferramenta.

A RST-IoT foi avaliada empiricamente por dois estudos. Como resultado, teve uma boa aceitação pelos engenheiros novatos, bem como evidências na sua utilidade e facilidade de uso de acordo com o estudo de observação.

Espera-se que com esses resultados, novas perspectivas de pesquisas possam ser exploradas em trabalhos futuros. Uma das oportunidades é o aprimoramento da RST-IoT, para apoiar um processo completo de especificação de requisitos para sistemas de software IoT, desde a especificação, até a validação. Outro ponto é a inserção do processo de inspeção de requisitos de sistemas de software IoT integrado a ferramenta Souza *et al.* (2019) [12]. Também, pode-se destacar o uso de verificação formal (*model checking*) de requisitos de software IoT utilizando *Machine Learning*.

### Agradecimentos

Os autores agradecem a todos os participantes de ambos os estudos experimentais. Além disso, também agradecemos a CALCOMP (sob nº de processo 10.242-8 – IIoT e Indústria 4.0). Este trabalho também foi parcialmente financiado pelo CNPq e CAPES - Código Financeiro 001.

### Referências

1. Pinon, M.M.B., et al. Applications and Advantages of the Internet of Things at Industry. *Journal of Engineering and Technology for Industry Applications*, v. 4, n. 15, p. 189-194, 2018.
2. Atzori, L., Iera, A., Morabito, G. The internet of things: A survey. *Computer networks*, v. 54, n. 15, p. 2787-2805, 2010.
3. Ur Rehman, T., Khan, M.N.A., Riaz, N. Analysis of requirement engineering processes, tools/techniques and methodologies. *International Journal of Information Technology and Computer Science*, v. 5, n. 3, p. 40, 2013.
4. IEEE, Recommended Practice for Software Requirements Specifications, in IEEE, vol., no., pp.1-40, 20 Oct. 1998, doi: 10.1109/IEEESTD.1998.88286.

5. Mahalank, S.N., Malagund, K.B., Banakar, R.M. Non functional requirement analysis in IoT based smart traffic management system. In: International Conference on Computing Communication Control and automation. IEEE, 2016. p. 1-6.
6. Martins, L.E.G., et al. TERASE: Template para Especificação de Requisitos de Ambiente em Sistemas Embarcados. In: WER. 2010.
7. Costa, B., Pires, P.F., Delicato, F.C. Specifying Functional Requirements and QoS Parameters for IoT Systems. In: IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing, 15th Intl Conf on Pervasive Intelligence and Computing, 3rd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress. IEEE, 2017. p. 407-414.
8. Reggio, G. A UML-based proposal for IoT system requirements specification. In: Proceedings of the 10th International Workshop on Modelling in Software Engineering. ACM, 2018. p. 9-16.
9. Da Silva, D.V., De Souza, B.P., Gonçalves, T.S., Travassos, G.H. Uma Tecnologia para Apoiar a Engenharia de Requisitos de Sistemas de Software IoT. In: 23rd Iberoamerican Conference on Software Engineering, 2020, Curitiba.
10. Lima, V.M.C., Neto, A.G.S.S., Emer, M.C.F.P. Investigação Experimental E Práticas Ágeis: Ameaças À Validade De Experimentos Envolvendo A Prática Ágil Programação em Par. Revista Electronica de Sistemas de Informacao, v. 13, n. 1, p. 1, 2014.
11. Carver, J; Jaccheri, L, Morasca, S, Shull, F. Issues in using students in empirical studies in software engineering education. 5th International Workshop on Enterprise Networking and Computing in Healthcare Industry, p. 239–249, 2003. DOI: 10.1109/METRIC.2003.1232471.
12. Souza, B.P., Motta, R.C., Travassos, G.H. The first version of SCENARIOTCHECK: A Checklist for IoT based Scenarios. In: XXXIII Brazilian Symposium on Software Engineering, 2019, Salvador/BA, p. 219-223.
13. Silva, V.M.: SCENARIoT Support for Scenario Specification of Internet of Things-Based Software Systems, Master Dissertation, UFRJ, Rio de Janeiro (2019). <https://www.cos.ufrj.br/index.php/pt-BR/publicacoes-pesquisa/details/15/2900>.
14. Liao, Y., Deschamps, F., Loures, E.F.R., Ramos, L.F.P. Past, present, and future of Industry 4.0 - a systematic literature review and research agenda proposal. International Journal of Production Research. 2017, DOI:10.1080/00207543.2017.1308576.
15. GLASS, R.L. Software runaways-Lessons learned from massive software project failures. 1998. Prentice Hall.