



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

**ARNALDO SANTOS QUAGLIATO**

**USABILIDADE ALÉM DA INTERFACE DO USUÁRIO: REENGENHARIA  
DOS RECURSOS PRINCIPAIS DA USINN MODELER**

**RUSSAS**

**2024**

ARNALDO SANTOS QUAGLIATO

USABILIDADE ALÉM DA INTERFACE DO USUÁRIO: REFATORAÇÃO DOS  
RECURSOS PRINCIPAIS DA USINN MODELER

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

Orientadora: Prof. Dra. Anna Beatriz dos Santos Marques.

RUSSAS

2024

Dados Internacionais de Catalogação na Publicação  
Universidade Federal do Ceará  
Sistema de Bibliotecas  
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

---

Q21u Quagliato, Arnaldo Santos.  
Usabilidade além da interface do usuário: reengenharia dos recursos principais da usinn modeler / Arnaldo Santos Quagliato. – 2024. 61 f.

Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Russas,  
Curso de Engenharia de Software, Russas, 2024.

Orientação: Profa. Dra. Anna Beatriz dos Santos Marques.

1. Reactflow. 2. Usabilidade. 3. Escalabilidade. 4. Manutenibilidade de Software. 5. Experiência do Desenvolvedor. I. Título.

CDD 005.1

---

ARNALDO SANTOS QUAGLIATO

USABILIDADE ALÉM DA INTERFACE DO USUÁRIO: REFATORAÇÃO DOS  
RECURSOS PRINCIPAIS DA USINN MODELER

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

Aprovada em: \_\_/\_\_/\_\_\_\_.

BANCA EXAMINADORA

---

Profa Dra Anna Beatriz dos Santos Marques  
(Orientadora)  
Universidade Federal do Ceará (UFC)

---

Prof. Dr. Prof Dr Alexandre Matos Arruda  
Universidade Federal do Ceará (UFC)

---

Prof. Msc. Alex Felipe Ferreira Costa  
Dataprev

Dedico este trabalho, a minha família em  
especial a minha mãe e irmã.

## **AGRADECIMENTOS**

À minha família, deixo o meu mais sincero agradecimento. Em especial, à minha mãe e minha irmã Cleida e Danyelle, que foram meu alicerce durante toda a caminhada. Sem o apoio incondicional, a força e o carinho de vocês, esta conquista não seria possível. Vocês foram primordiais para que eu pudesse alcançar o ensino superior e concluir essa etapa tão importante da minha vida.

Gostaria de agradecer aos meus amigos os quais pude conhecer nessa trajetória pelo Vale do Jaguaribe. Aos meus amigos do Santiarlon, Alex Alan e Warton Cabral pelo apoio acadêmico e pessoal durante esse período. Também não poderia deixar de lembrar dos meus amigos em Fortaleza que me apoiaram desde o resultado para o SISU até os últimos trabalhos na graduação, Cassiano Santos, Letícia Leitão, Vitória Almeida, Guilherme Mafezoli, Gabriel Machado e Aiwass Alencar. Sou grato pela vida de todos.

Por fim, agradeço à Universidade Federal do Ceará, pela excelente estrutura de ensino e a todo o corpo docente que compartilhou seus ensinamentos e experiências ao longo da minha jornada acadêmica. Em especial, ao Professor Alexandre Arruda, cuja orientação foi determinante nas minhas decisões acadêmicas e profissionais. Minha gratidão também se estende à Professora e Coordenadora do LUDI, Anna Beatriz, por sua paciência, incentivo e orientação. Sua liderança foi fundamental para a realização deste trabalho e para meu desenvolvimento como profissional.

Aos professores participantes da banca examinadora obrigado pelo tempo, pelas valiosas colaborações e sugestões.

Scientia liberet

## RESUMO

Este trabalho tem como objetivo realizar uma refatoração dos recursos principais da ferramenta USINN Modeler, utilizada para criar e editar diagramas baseados na notação USINN, a qual visa facilitar a modelagem de usabilidade em sistemas interativos. Com o objetivo de melhorar a experiência do desenvolvedor e possibilitar a evolução da ferramenta, o problema investigado gira em torno da usabilidade e da necessidade de atualizar as tecnologias da ferramenta, visando torná-la mais acessível e escalável. A pesquisa foca na melhoria da experiência do desenvolvedor e na usabilidade da ferramenta. Para alcançar esses objetivos, foi adotada a metodologia qualitativa, com a coleta de dados por meio de entrevistas com usuários da ferramenta, análise sobre usabilidade de trabalhos anteriores e artefatos sobre a ferramenta. A refatoração incluiu a substituição da biblioteca mxgraph pelo React Flow, visando maior manutenibilidade e clareza no código. Os resultados indicam melhorias na produtividade dos desenvolvedores, além de apontarem novas direções para testes com um público mais amplo e a continuidade do desenvolvimento.

**Palavras-chave:** usinn modeler; usabilidade; experiência do desenvolvedor; refatoração; reactflow.



## ABSTRACT

This work aims to refactor the core features of the USINN Modeler tool, used to create and edit diagrams based on the USINN notation, which is designed to facilitate usability modeling in interactive systems. In order to improve the developer's experience and enable the tool's evolution, the research problem revolves around usability and the need to update the tool's technologies, making it more accessible and scalable. The research focuses on enhancing both the developer experience and the tool's usability. To achieve these objectives, a qualitative methodology was adopted, with data collection through user interviews, usability analysis from previous studies, and review of artifacts related to the tool. The refactoring included replacing the mxgraph library with React Flow, aiming for greater maintainability and code clarity. Preliminary results indicate improvements in developer productivity, as well as highlighting new directions for broader testing and continued development.

**Keywords:** usinn modeler; usability; developer experience; refactoring; react flow.

## LISTA DE FIGURAS

Figura - 1	Notações da USINN.....	20
Figura - 2	Exemplo de diagrama USINN sendo elaborado com o USINN Modeler .....	23
Figura - 3	Processo de design adotado pelo time .....	28
Figura - 4	Etapas da metodologia adotada .....	30
Figura - 5	Comparação entre mxgraph e reactflow .....	35
Figura - 6	Trecho de código mxgraph .....	36
Figura - 7	Trecho de código refatorado .....	37
Figura - 8	Regras USINN .....	38
Figura - 9	Protótipo desenvolvido no Figma .....	39
Figura - 10	Estrutura de pastas da refatoração.....	44
Figura - 11	Ação ao clicar para selecionar conexão .....	45
Figura - 12	Ação ao clicar para selecionar conexão .....	45
Figura - 13	Estrutura base reactflow.....	46
Figura - 14	Estrutura de dados de um nó .....	47
Figura - 15	Estrutura de dados de uma aresta.....	47
Figura - 16	Exemplo da notação após a refatoração.....	48
Figura - 17	Regra contendo validação entre conexões.....	49
Figura - 18	Menu refatorado.....	51
Figura - 19	Menu atual .....	51
Figura - 20	Diagrama teste para exporta .....	52
Figura - 21	Resultado do diagrama exportado.....	52
Figura - 22	Diagrama de teste para exportar .....	53
Figura - 23	Resultado do diagrama exportado do diagrama refatorado .....	54
Figura - 24	Unidade de apresentação sobrepondo elemento .....	55
Figura - 25	Unidade de apresentação não arrastando elementos juntos .....	55
Figura - 26	Unidade de apresentação refatorada antes de ser arrastada .....	55
Figura - 27	Unidade de apresentação refatorada após ser arrastada.....	56
Figura - 28	Gráfico e pergunta um sobre experiência com o código refatorado .....	56
Figura - 29	Gráfico e pergunta dois sobre o código refatorado .....	57
Figura - 30	Gráfico e pergunta três sobre o código refatorado.....	57
Figura - 31	Gráfico e pergunta quatro sobre o código refatorado .....	57

## **LISTA DE TABELAS**

Tabela 1 - Resultado da análise qualitativa. ....	24
---	----

## **LISTA DE ABREVIATURAS E SIGLAS**

DX	Experiência do desenvolvedor
ISO	Organização internacional para padronização
PEOU	Facilidade de Uso Percebida
TAM	Modelo de aceitação de tecnologia
UI	User Interface
US	História do usuário
USINN	USability-oriented INteraction and Navigation Mo
UX	Experiência do usuário

## SUMÁRIO

<b>1 INTRODUÇÃO .....</b>	<b>15</b>
<b>2 OBJETIVOS .....</b>	<b>17</b>
2.1 Objetivo geral .....	17
2.2 Objetivos específicos .....	17
<b>3 FUNDAMENTAÇÃO TEÓRICA.....</b>	<b>18</b>
3.1 Usabilidade .....	18
3.2 USINN .....	19
3.3 USINN Modeler .....	21
3.4 Análise qualitativa .....	23
3.5 Refatoração .....	24
3.6 Experiência do desenvolvedor .....	25
<b>4 TRABALHOS RELACIONADOS .....</b>	<b>26</b>
4.1 Estruturação e apresentação de sistemas de ajuda on-line para software educacional .....	26
4.2 Percepções de Professores sobre Recursos de Ajuda em Ferramentas de Programação baseadas em Blocos .....	27
4.3 Design de interação com a notação USINN: resultados iniciais sobre a experiência prática de designers .....	28
4.4 Exploring novice designers acceptance of the use of a tool for modeling the interaction and usability of information systems.....	28
<b>5 PROCEDIMENTOS METODOLÓGICOS.....</b>	<b>30</b>
5.1 Estudo sobre a notação da USINN e ferramenta USINN Modeler .....	30
5.2 Planejamento das entrevistas .....	31
5.3 Condução das Entrevistas .....	32
5.4 Análise qualitativa dos dados .....	33
5.5 Design e desenvolvimento .....	34
<b>6 RESULTADOS.....</b>	<b>40</b>
6.1 Entrevista e análise qualitativa .....	40
6.2 Refatoração da USIN Modeler .....	43

6.3 Avaliações Preliminares .....	50
<b>7 CONCLUSÃO E TRABALHOS FUTUROS.....</b>	<b>59</b>
<b>REFERÊNCIAS .....</b>	<b>60</b>

## 1 INTRODUÇÃO

O objetivo do design e avaliação de sistemas, produtos e serviços para usabilidade é habilitar usuários a alcançarem seus objetivos de forma eficiente e satisfatória, levando em consideração o contexto de uso. Além disso, deve-se considerar que usabilidade é um conceito mais amplo do que o senso comum "fácil de usar" ou "interface amigável" (ISO, 9241-11, 2018).

A ISO considera relevantes aspectos como a minimização de riscos e consequências indesejáveis de erros durante a utilização, bem como a manutenção, medida em que permite a realização eficaz, eficiente e satisfatória das tarefas de manutenção. Portanto, é crucial compreender que a usabilidade também impacta na capacidade de desenvolver um projeto sustentável, visto do ponto de vista do desenvolvedor o qual é responsável pelo desenvolvimento contínuo.

Ainda sobre usabilidade, Nielsen (1994) apresenta algumas afirmações. Por exemplo, os arquivos de log de sequências de interação do usuário geralmente ajudam mostrando onde o usuário pausou ou perdeu tempo, e quais erros foram encontrados com mais frequência.

O autor também faz observações sobre a consistência de um produto, concluindo que os usuários não deveriam se perguntar se palavras, situações ou ações diferentes significam a mesma coisa. Em corroboração a isso, Perlman (1989), descreve que a consistência deve ser aplicada em todas as diferentes mídias que compõem a interface total do usuário, incluindo não apenas as telas do aplicativo, mas também a documentação, o sistema de ajuda on-line e quaisquer tutoriais on-line ou gravados em vídeo.

Assim, este trabalho se propõe a entender e melhorar a usabilidade da ferramenta USINN Modeler. Para esse fim, foi realizada uma análise qualitativa como estratégia para apoiar o aprendizado e a adoção da USINN Modeler, na qual será desenvolvido um roteiro de entrevista e reuniões gravadas para o recolhimento de dados por parte do público-alvo. O objetivo é conduzir um estudo que possa guiar o processo de desenvolvimento de novos recursos e melhorias para os recursos existentes.

A ferramenta USINN Modeler nasce com a finalidade de tornar mais prática e sintaticamente correta a criação e edição de diagramas USINN (COSTA; MARQUES, 2019). A notação USINN foi desenvolvida para facilitar a representação de mecanismos de usabilidade em atividades de modelagem, permitindo a ampliação de elementos de navegação e interação correlacionados com a usabilidade (MARQUES; BARBOSA; CONTE, 2017).

E a busca contínua por soluções tecnológicas inovadoras tem moldado um cenário em constante evolução no mundo contemporâneo. Nesse contexto, a ferramenta USINN Modeler se destaca como uma promissora contribuição para otimizar o entendimento de regras de negócio e estudo da área de Interação Humano Computador (IHC) por meio da modelagem de interação e navegação.

Corroborando esse cenário de inovação contínua, pode-se observar na pesquisa da Marques (2019) que designers mais experientes tiveram uma percepção positiva, especialmente em relação à modelagem de interação e navegação, simplificando o processo de prototipação. Além disso, também foram percebidos aspectos positivos para pessoas sem experiência, destacando-se a facilidade para desenvolvimento de modelos, uma perspectiva importante principalmente para iniciantes na computação.

Assim, com objetivo de apoiar a evolução do projeto, esse trabalho irá demonstrar o processo de transição da biblioteca descontinuada mxgraph a qual foi atualizada para reactflow, essa mudança busca alcançar um código modularizado facilitando a manutenção por parte dos desenvolvedores, melhorias na interface da aplicação, validação e inclusão de regras ao sistema.

O restante deste documento está organizado da seguinte maneira: na Seção 2 são apresentados os objetivos de pesquisa. A Seção 3 descreve a fundamentação teórica para o entendimento e realização deste trabalho. A Seção 4 apresenta os trabalhos relacionados. Na Seção 5 são apresentados os procedimentos metodológicos adotados para a execução deste trabalho. A Seção 6 apresenta a análise dos resultados. Finalmente, na Seção 7 apresenta a conclusão, nas quais também serão abordados trabalhos futuros.



## **2 OBJETIVOS**

Nesta seção, são descritos o objetivo geral e os objetivos específicos desta pesquisa.

### **2.1 Objetivo geral**

O objetivo central deste trabalho é melhorar a interação do usuário durante o uso da plataforma, tornando o processo de aprendizado e adoção da ferramenta mais eficaz e acessível para o público-alvo da USINN Modeler.

### **2.2 Objetivos específicos**

- 2.2.1 Identificar os principais desafios enfrentados pelos usuários ao interagirem com a ferramenta USINN Modeler, com foco na usabilidade e na navegação do sistema.
- 2.2.2 Compreender e desenvolver recomendações e soluções práticas para melhorar a usabilidade e a navegabilidade da ferramenta. De forma que a implementação de melhorias refatoração dos recursos de diagramação presentes, visem a experiência do desenvolvedor durante o processo de manutenção e implementação de novas funcionalidades
- 2.2.3 Selecionar tecnologias open-source com documentação robusta para desenvolvimento de diagramas.
- 2.2.4 Adaptar a ferramenta de maneira a atender às regras de negócio e às necessidades da solução.
- 2.2.5 Coletar evidências iniciais sobre a melhoria da usabilidade e qualidade da solução desenvolvida.

### 3 FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta os conceitos e teorias necessários para a compreensão deste trabalho. No tópico 3.1 são apresentados conceitos de usabilidade e sua relevância para esse trabalho, já no tópico 3.2 são apresentadas informações sobre a notação USINN. O tópico 3.3 aborda a ferramenta USINN Modeler, enquanto o tópico 3.4 fala sobre análise qualitativa, método de extrema relevância para o desenvolvimento do presente trabalho, já o tópico 3.5 refatoração representa a principal atividade desempenhada durante a etapa de desenvolvimento e o tópico 3.6 trata da experiência do desenvolvedor e busca representar um dos objetivos a serem atingidos durante a etapa de desenvolvimento.

#### 3.1 Usabilidade

Usabilidade é o termo técnico usado para descrever a qualidade de uso de uma interface (Bevan, 1995). Portanto, o entendimento dos recursos de ajuda, que é de grande valia para o usuário, será entendido por meio de entrevista, visando compreender as principais dificuldades enfrentadas pelos usuários ao final da pesquisa.

A usabilidade é um conceito essencial no design de produtos e sistemas interativos, pois está intrinsecamente relacionada à eficácia e à satisfação do usuário. Segundo Nielsen (1993), a usabilidade é a medida em que um produto pode ser utilizado por usuários específicos com eficácia, eficiência e satisfação em um contexto de uso específico. Isso implica que um sistema ou produto deve ser fácil de aprender, eficiente de usar e proporcionar uma experiência satisfatória aos seus usuários.

De acordo com Shneiderman (1998), a usabilidade é composta por cinco qualidades-chave: eficácia, eficiência, aprendizado, memorização e satisfação. A eficácia refere-se à capacidade do usuário alcançar seus objetivos com sucesso. A eficiência diz respeito à relação entre o esforço do usuário e a realização de tarefas. O aprendizado relaciona-se com a facilidade com que novos usuários podem começar a usar o sistema, aspecto central para o trabalho atual. A memorização refere-se à facilidade de lembrar como realizar tarefas após um período de inatividade. A satisfação, por sua vez, está relacionada à experiência global do usuário ao interagir com o sistema.

Para garantir a usabilidade de um produto ou sistema, é fundamental que durante o desenvolvimento de funcionalidades sejam considerados os conceitos citados anteriormente desde as fases iniciais. Nesse sentido, a abordagem centrada no usuário, com coleta de feedback, desempenha um papel crucial na melhoria contínua da usabilidade do produto. Portanto,

designers e desenvolvedores devem esforçar-se para seguir essas diretrizes a fim de criar um produto mais eficaz, eficiente e satisfatório.

### 3.2 USINN (USability-oriented Interaction and Navigation Model)

Antes de entender melhor sobre USINN, é válido ressaltar sobre modelos de interação. Segundo Simone (2003), um modelo de interação descreve a comunicação entre o usuário e o sistema, especificando quando o usuário pode realizar tarefas específicas para atingir determinados objetivos, quando o usuário pode selecionar ou especificar algumas entradas de dados, e quando o sistema pode processar a informação do usuário e apresentar o conteúdo e feedback adequado. Já os modelos de navegação representam nós de navegação e fluxos de navegação entre eles.

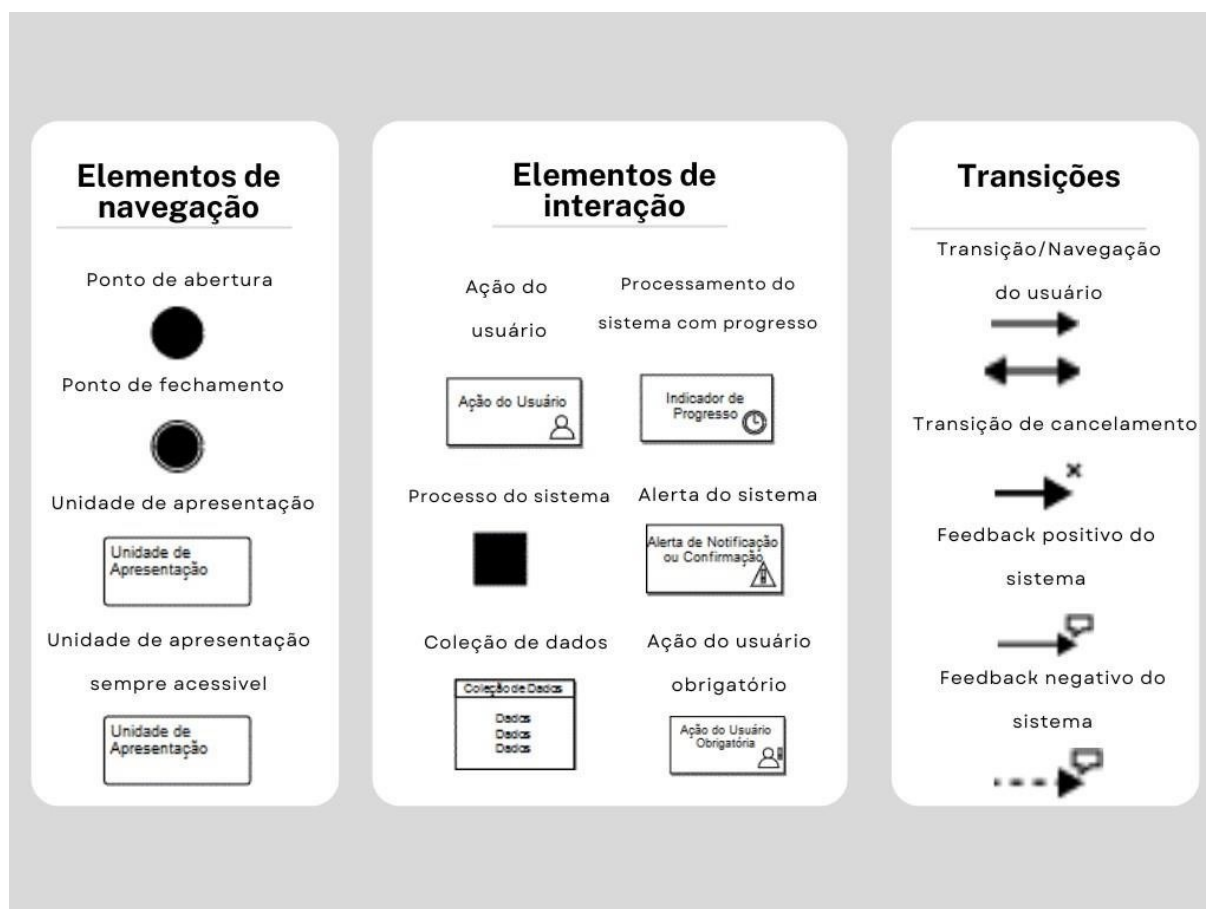
Com o objetivo de representar graficamente a interação e navegação de sistemas interativos, considerando aspectos de usabilidade, foi proposto o modelo denominado de USINN (*USability-oriented Interaction and Navigation Model*) para proporcionar melhoria na usabilidade e qualidade de uso no produto final (MARQUES *et al.*, 2016).

Dessa forma, pode-se definir USINN como uma ferramenta para modelar a interação e a navegação de sistemas interativos com foco na usabilidade. Assim, dentre seus objetivos destaca-se:

- **Navegação do usuário:** permite representar unidades de apresentação de interface e seus relacionamentos.
- **Interação do usuário:** permite representar ações do usuário e respostas do sistema que ocorrem nas unidades de apresentação.
- **Representa a usabilidade:** permite considerar aspectos de usabilidade que podem ser incorporados nas soluções alternativas.

É importante entender as notações envolvidas, que serão demonstradas a seguir.

Figura - 1 Notações da USINN



Fonte: Elaborado pelo autor.

Ademais, uma breve explicação sobre a notação:

- **Ponto de abertura e ponto de encerramento:** indicam o início e término da navegação/interação do usuário com o sistema, respectivamente.
- **Unidade de apresentação:** é a base da estrutura navegacional do sistema, restringindo a interação que um usuário pode realizar pela interface a cada momento.
- **Unidade de apresentação sempre acessível:** é uma unidade de apresentação que deve estar sempre acessível durante a interação.
- **Ação do usuário:** descreve ações que o usuário pode realizar para alcançar um determinado objetivo de interação.
- **Ação do usuário obrigatória:** descreve ações do usuário obrigatórias para que o usuário prossiga em direção a um objetivo de interação.

- **Coleção de dados:** elemento que contém os dados apresentados e utilizados durante as operações do usuário. As preferências e elementos favoritos do usuário podem ser armazenados nas coleções de dados.
- **Processo do sistema:** representa um processamento interno do sistema após uma solicitação do usuário. Após sua conclusão, o sistema deve fornecer feedback ao usuário.
- **Processo do sistema com indicador de progresso:** representa um processamento interno do sistema que informa ao usuário sobre o seu andamento, por meio de um indicador de progresso.
- **Alerta de notificação:** representa um alerta que o sistema pode emitir durante a interação do usuário com o sistema, devido a determinadas condições. O alerta de notificação não irá interromper a interação do usuário, nem requerer uma resposta do usuário.
- **Navegação:** relacionamento entre unidades de apresentação da interface do sistema.
- **Transição do usuário:** diálogo em que o usuário escolhe como prosseguir a interação entre um conjunto de ações fornecidas pelo sistema.
- **Transição de cancelamento:** descreve a possibilidade de o usuário desfazer ou cancelar ações.
- **Feedback do sistema:** resposta fornecida pelo sistema a uma requisição do usuário. Pode ser um feedback de sucesso ou de falha.
- **Query:** relacionamentos necessários para demonstrar que está buscando informações em uma coleção de dados.

### 3.3 USINN Modeler

Permanecer com modelos de interação e navegação consistentes não é uma tarefa fácil, uma vez que os modelos de interação utilizam diferentes elementos como base, enquanto os modelos de navegação geralmente se fundamentam em unidades de apresentação da interface. Uma possível solução é utilizar elementos comuns em ambos (MARQUES *et al.*, 2016).

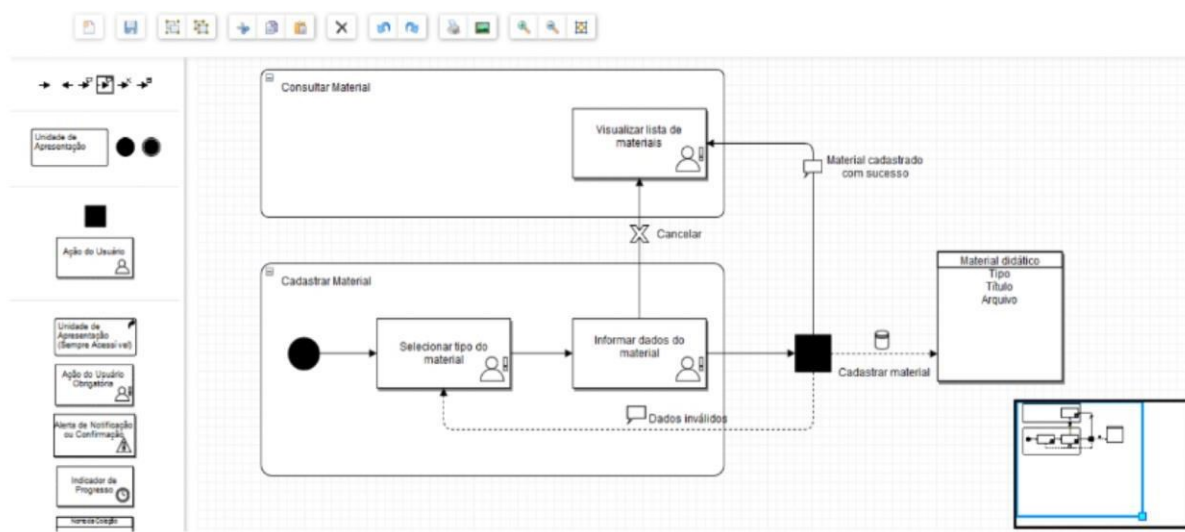
No entanto, elaborar diagramas utilizando USINN não era algo prático devido à falta de uma ferramenta adequada para criar esses diagramas com esta notação, o que os tornava suscetíveis a erro de sintaxe. Com o propósito de facilitar a criação e edição de modelos nesta notação, foi desenvolvida uma ferramenta web para elaboração desses modelos (COSTA; MARQUES, 2019).

As ferramentas de diagramação oferecem uma variedade de recursos que permitem a disposição de elementos gráficos, como texto, imagens e gráficos, de forma esteticamente atraente e organizada. Essas ferramentas geralmente fornecem modelos predefinidos, opções de personalização, e são amplamente utilizadas na produção de diagramas UML. Entretanto, representar um fluxo completo de interações entre o usuário e o sistema pode ser uma tarefa complexa.

Por isso, a ferramenta denominada de *USINN Modeler* torna a elaboração de modelos USINN mais prática e sintaticamente correta. Sua primeira versão inclui alguns dos requisitos essenciais para a notação USINN. Costa e Marques (2019) apresentam as principais funcionalidades do *USINN Modeler*:

- **Gerenciar graficamente os elementos da notação:** através da técnica “arrastar e soltar” é possível inserir elementos na notação.
- **Conectar os elementos da notação:** a partir dos relacionamentos permitidos entre os elementos, é possível inserir conectores entre eles.
- **Impedir conexões inválidas:** há uma verificação para identificar se a conexão é válida, ao conectar um elemento a outro.
- **Validar sintaxe do diagrama:** cada ação é avaliada dentro das regras do USINN, caso não seja permitida, é exibida uma mensagem de alerta.

Figura - 2 Exemplo de diagrama USINN sendo elaborado com o USINN Modeler



Fonte: Costa e Marques (2019).

Para um melhor entendimento sobre a notação, seguem algumas explicações sobre o diagrama acima. Os quadrantes com título "Consultar Material" e "Cadastrar Material" representam os objetivos da interação, e os objetos contidos neles representam as ações necessárias para alcançar esses objetivos.

No cartão "Visualizar lista de materiais", "Selecionar tipo do material" e "Informar dados do material", temos ações obrigatórias que o usuário deve realizar para concluir o propósito da sua interação. Já "Dados Inválidos" e "Material Cadastrado com Sucesso" são feedbacks retornados para o usuário, sendo representado por linha pontilhada em caso de insucesso e a seta comum seria com sucesso.

As notações de círculo preto e quadrado preto representam, respectivamente, o início da interação e um momento de processamento. Nesse caso seria o processo de cadastrar o material no banco de dados, que está ligado a "Material didático" a qual representa a entidade no sistema, sendo "Título", "Tipo" e "Arquivo" as informações necessárias para a escrita no banco de dados.

### 3.4 Análise Qualitativa

Segundo Marli (1983), a análise qualitativa apreende o caráter multidimensional dos fenômenos em sua manifestação natural, bem como captar os diferentes significados de uma experiência prática, auxiliando a compreensão do indivíduo no seu contexto. Consiste em um

conjunto de práticas interpretativas e materiais que permitem descrever seres humanos, suas atividades e contexto em que são observados.

A pesquisa qualitativa gera uma série de representações. No caso deste trabalho, foram geradas notas de campo, anotações de respostas de entrevistas, conversas, gravações e transcrições das entrevistas, que serão usados na análise e na interpretação dos dados. Dentre as abordagens que podem ser utilizadas:

- **Análise temática:** Procura padrões no significado dos dados para encontrar temas. Relevante para extrair temas em áreas pouco exploradas.
- **Análise de conteúdo:** Para identificar os padrões que emergem do texto, agrupando o conteúdo em palavras, conceitos e temas.
- **Teoria fundamentada:** Focada em descobrir ou construir teorias ou modelos a partir de dados analisados por comparações constantes.
- **Análise do discurso:** Analisar como os dados possuem sentidos e significados ao se relacionar com outros dados sobre um determinado contexto social.
- **Análise narrativa:** Concentra-se nas histórias que as pessoas contam e na linguagem que usam para entendê-las. Analisar a estrutura, personagens e eventos presentes.

Também é válido ressaltar algumas terminologias tais como código que é uma palavra ou frase curta a qual representa uma ideia, agrupando vários trechos de um texto. Já codificação é a ação de identificar os trechos de um texto com uma ideia em comum. As abordagens para codificação podem ser indutivas (abertas), onde o código e suas definições emergem a partir da análise do artefato, e dedutiva (fechada), a qual antes de iniciar a análise as definições já são estabelecidas.

### 3.5 Refatoração

A refatoração é o processo de reestruturar o código-fonte de um software, visando melhorar sua estrutura interna sem alterar seu comportamento externo observável (Fowler, 1999). Essa prática é fundamental para manter a qualidade do software ao longo do tempo, tornando-o mais legível, modular e fácil de manter. Segundo Fowler (1999), a refatoração permite que os desenvolvedores melhorem o design do código existente, reduzindo a complexidade e facilitando a detecção e correção de erros.



Beck (2000) reforça que a refatoração é um elemento-chave nas metodologias ágeis, permitindo que o design do software evolua continuamente em resposta às mudanças nos requisitos. Práticas comuns de refatoração incluem a extração de métodos, renomeação de variáveis e classes, simplificação de expressões condicionais e redução de código duplicado (FOWLER, 1999). Martin (2009) enfatiza que um código limpo é essencial para a produtividade dos desenvolvedores, e a refatoração é a ferramenta que permite alcançar esse objetivo.

No contexto deste trabalho, a refatoração dos recursos principais da USINN Modeler é essencial para melhorar a experiência dos desenvolvedores durante o processo de manutenção e implementação de novas funcionalidades. Ao aprimorar a estrutura interna do código, torna-se mais fácil compreender, modificar e estender a ferramenta, garantindo sua evolução contínua e qualidade.

### **3.6 Experiência do desenvolvedor**

A experiência do desenvolvedor é um aspecto crucial no desenvolvimento de software, referindo-se à qualidade das interações dos desenvolvedores com as ferramentas, tecnologias e processos envolvidos em seu trabalho diário. As escolhas técnicas realizadas durante o desenvolvimento têm um impacto direto na produtividade, satisfação e eficiência das equipes (NUCCI *et al.*, 2020). Vale ressaltar que a pesquisa aponta a relação entre bugs encontrados e qualidade do código de forma que selecionar tecnologias apropriadas, linguagens de programação adequadas, frameworks eficientes e ferramentas de apoio contribui significativamente para facilitar o trabalho dos desenvolvedores.

Segundo Graziotin, ao investigar a felicidade do desenvolvedor perceberam que dentre os fatores de descontentamento estariam associados a ficar preso na solução do problema, má qualidade de código e dificuldade para depurar o código. De forma a correlacionar a produtividade com a satisfação durante o desenvolvimento (Graziotin, 2017).

Portanto, no contexto do USINN Modeler, focar na experiência do desenvolvedor é essencial para tornar o processo de desenvolvimento mais eficiente e agradável. A escolha de bibliotecas e frameworks atualizados e amplamente suportados, como bibliotecas modernas de diagramação, pode melhorar significativamente a produtividade dos desenvolvedores que trabalham na ferramenta.

## **4 TRABALHOS RELACIONADOS**

Os trabalhos apresentados nesta seção estão relacionados à estruturação e apresentação de recursos de ajuda, sobre a notação USINN e a plataforma USINN Modeler. As duas primeiras seções irão descrever os recursos oferecidos por outras ferramentas no ecossistema educacional. E a última estaria pautada em compreender a percepção da ferramenta em trabalhos anteriores.

### **4.1 Estruturação e apresentação de sistemas de ajuda on-line para software educacional**

Pelissari (2009) apresenta uma revisão bibliográfica de trabalhos relacionados à área de sistemas de ajuda on-line em geral. Para esta pesquisa foi utilizado o processo de Revisão Sistemática. Ao final desta revisão sistemática, foram analisados 19 artigos, com publicações posteriores ao ano de 2003.

De início, foi realizada uma análise sobre os tipos de recursos de ajuda existentes, levando em consideração seus conteúdos, potencialidades e problemas. Em seguida, a fim de verificar se os sistemas de ajuda existentes realmente auxiliam seus usuários e o que eles gostariam que fosse contemplado nestes sistemas, foram realizadas entrevistas. Os próximos passos foram elaborar um teste com usuários, que foi aplicado a potenciais usuários de software educacional (alunos, professores e coordenadores de laboratório de informática) e por fim, a partir dos estudos teóricos e da coleta de dados foi elaborada uma proposta de estruturação e apresentação dos conteúdos de sistemas de ajuda on-line para software educacional, a qual foi explorada através da criação de um protótipo.

Dessa forma, é válido ressaltar os diferentes conceitos de recursos de ajuda e como a literatura os divide. Para Rosenbaum, Kantner e Dworman (2005), os conceitos serão divididos em processual (informações que descrevem a forma de realizar uma tarefa), interface (instruções sobre a interface e sobre como utilizar cada um de seus elementos), domínio de informação (informação sobre os recursos do software), suplementar (informação sobre termos técnicos, como em um glossário), e meta-instruções (informações sobre o uso do software). Já Ascencio (2000), classifica os tipos de conteúdo de ajuda da seguinte forma: orientada ao objetivo (“O que eu posso fazer com este programa?”), descritiva (“O que é isto? Para que serve isto?”), procedimento (“Como eu faço isso?”), interpretativa (“Porque e como isso aconteceu?”) e navegação (“Onde eu estou?”). Essa classificação foi estendida por Herrmann e Silveira

(2005) que incluíram a ajuda de exemplo (que sugere a apresentação de exemplos sobre como usar o sistema). Huang, Lu e Twidale (2005), Voulligny e Robert (2005) e Herzog e Shahmehri (2007) apresentam diferentes mecanismos de apresentação dos conteúdos da ajuda, sendo os mais encontrados: ajuda contextual (instruções visíveis em um painel de ajuda ou a um clique de distância do atual estado do usuário, com informações sobre o tópico em questão, sem perder o estado atual da apresentação); ajuda local (dicas que aparecem quando o usuário detém temporariamente o mouse sobre um elemento da interface (ajuda flutuante)).

Assim, conhecer os conceitos apresentados pelos autores acima grande valia para a construção do roteiro da entrevista para identificar quais recursos os usuários da USINN gostariam que fossem implementados.

#### **4.2 Percepções de Professores sobre Recursos de Ajuda em Ferramentas de Programação baseadas em Blocos**

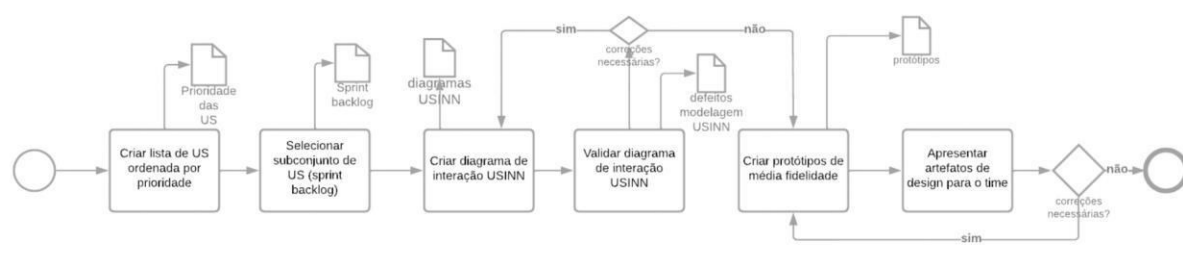
Brezolin (2022) irá apresentar um estudo sobre software educacional, em particular, questões relacionadas a ensino e aprendizagem em ferramentas de programação baseada em blocos. Neste documento foi abordado principalmente a visão do professor, de forma que se utilizou a metodologia de caráter exploratório a qual foram coletadas informações por meio de questionários respondidos por 95 participantes.

Sobre esse trabalho é válido ressaltar que devido ao caráter exploratório é possível encontrar muitas perspectivas, principalmente do ponto de vista de como os usuários buscam as informações, ficando claro a necessidade do uso do Youtube, plataforma a qual boa parte dos entrevistados afirmam buscar informações. Além disso, é válido ressaltar que o estudo deixa claro algumas preferências por parte do usuário que podem ser aplicados para o presente trabalho, como a ajuda visual, haja vista que 69.7% na pesquisa tem preferência por vídeo e 10.6% por imagem. Também se percebe que além dos vídeos, exemplos de uso, Perguntas Realizadas com Frequência (FAQ) são ferramentas bem avaliadas por parte dos professores que responderam ao questionário e que podem ser implementadas para a USINN.

### 4.3 Design de interação com a notação USINN: resultados iniciais sobre a experiência prática de designers

Feitosa (2022) apresenta um estudo realizado com designers sobre a notação USINN. Durante a investigação, foram criadas 20 histórias de usuários (US) e mecanismos de usabilidade a serem atendidos em cada US. Em seguida, uma vez que os diagramas desenvolvidos foram validados pela equipe era iniciado o processo de prototipação de alta fidelidade. O esquema a seguir irá demonstrar o fluxo de trabalho seguido.

Figura - 3 Processo de design adotado pelo time



Fonte: Feitosa e Victor (2022).

Ao fim, a percepção de dois designers foi extraída por meio de duas perguntas: "Você acredita que conseguiu atingir seu objetivo de modelagem?", " Você sentiu de alguma forma contribuiu para a evolução do trabalho do time?". Dessa forma, em relação à eficácia na modelagem, P1 (participante) mencionou que "Em alguns casos senti que o objetivo foi concluído, em outros casos eu me senti perdido e não consegui identificar se realmente o meu objetivo foi atingido ou então não sabia como proceder". Assim, parte da problemática dos usuários da USINN está no entendimento da notação e a certeza de que estão acertando no uso da notação e para o desenvolvimento de ferramentas de ajuda é importante levar esse ponto em consideração.

### 4.4 Exploring novice designers acceptance of the use of a tool for modeling the interaction and usability of information systems

Esse artigo tem como objetivo avaliar o processo de transferência tecnológica da ferramenta para empresas de desenvolvimento de software e, portanto, é importante assegurar que com a sua adoção as empresas irão se manter competitivas e que a ferramenta agregará valor para o seu processo de desenvolvimento. Assim, o estudo conduziu uma avaliação com

usuários com base no modelo de aceitação de tecnologia (TAM - Technology Acceptance Model) [23], que avalia dimensões que impactam na aceitação de uma tecnologia em um contexto específico.

A abordagem foi focada em compreender os interesses do usuário final e para esse estudo foram considerados como designers de interação novatos por serem estudantes de graduação com conhecimento em usabilidade, mas sem experiência profissional na área. A avaliação foi desenvolvida com os usuários em laboratório utilizando ferramenta Google Formulário, as questões investigadas foram "*Qual a percepção dos usuários sobre o uso da ferramenta USINN Modeler para a modelagem de interação?*" e "*Quais as oportunidades de melhorias da ferramenta USINN Modeler?*", com um total de 25 participantes e 22 respostas consideradas.

Em conclusão a pesquisa revelou uma clara percepção da utilidade da ferramenta para modelagem de interações e navegação. No entanto, quanto à usabilidade percebida, uma proporção significativa de participantes demonstrou neutralidade ou discordância, apontando para áreas de melhoria, especialmente relacionadas à Facilidade de Uso Percebida (PEOU). Dentre elas, foi possível identificar a necessidade de melhorar a navegação da página de criação de diagramas. Assim, entende-se que melhorias precisam ser desenvolvidas e a fim de habilitar a evolução da ferramenta deve ser necessário a refatoração, incluindo a possibilidade do uso de novas ferramentas para diagramação que facilitem a evolução do projeto.

## 5 PROCEDIMENTOS METODOLÓGICOS

Este capítulo apresenta as atividades que foram realizadas para o alcance dos objetivos deste trabalho. A Figura 4 apresenta as etapas do processo metodológico definido para o desenvolvimento deste trabalho.

Figura - 4 Etapas da metodologia adotada



Fonte: Elaborado pelo autor (2023)

### 5.1 Estudo sobre a notação da USINN e ferramenta USINN Modeler

Esta etapa teve a finalidade de realizar um estudo de temas que estão relacionados a este trabalho, sendo possível um melhor entendimento do projeto e a definição do escopo a qual estaria atuando.

Segundo Goldenberg (2002) ele conceitua que **brainstorming** seria um grupo de indivíduos que apresenta ideias sem nenhuma ordem específica e nenhum critério para julgamento de seus méritos é aplicado. Cada ideia é considerada boa e quanto mais diversas forem as ideias, melhor. Nesse contexto, em reunião direta com orientadora durante foi identificado as necessidades para o desenvolvimento do projeto e alinhado ao conhecimento de mercado e portanto seria interessante o desenvolvimento de ferramentas de ajuda que pudessem melhorar o engajamento e entendimento com a USINN Modeler.

Nesse período foi possível entender sobre o modelo de interação e navegação orientado à usabilidade (USINN) de forma que além da apresentação do projeto foi realizada a leitura de trabalhos anteriores. Entre eles o material "Projetando a interação e navegação de sistemas interativos com foco na usabilidade", aula na qual a Dra. Anna Beatriz Marques, autora da notação USINN, pode explicar melhor sobre os conceitos da ferramenta e como utilizar sua notação.

Outro método adotado para compreender as funcionalidades e estágio atual do desenvolvimento da ferramenta USINN Modeler foi a etnografia. Segundo Sommerville (2011)

etnografia é uma técnica de observação que pode ser usada para compreender os processos operacionais e ajudar a extrair os requisitos de apoio para esses processos. Um analista faz uma imersão no ambiente de trabalho em que o sistema será usado. O trabalho do dia a dia é observado e são feitas anotações sobre as tarefas reais em que os participantes estão envolvidos.

O valor da etnografia é que ela ajuda a descobrir requisitos implícitos do sistema que refletem as formas reais com que as pessoas trabalham, em vez de refletir processos formais definidos pela organização. Assim, a participação do autor deste trabalho em reuniões com os integrantes do time de desenvolvimento da USINN Modeler foi de grande valia, pois foi possível compreender sobre a ferramenta, notação, backlog de desenvolvimento e os objetivos da equipe com o projeto.

## 5.2 Planejamento das entrevistas

A fim de ter uma maior assertividade quanto aos recursos de ajuda necessários para a USINN Modeler foi estabelecido um roteiro, o qual foi aprovado pela orientadora. Além disso, foi definido o público-alvo que iria ser entrevistado como sendo professores e alunos que já tiveram contato com a ferramenta. Ademais, foi utilizado o *Google Forms* para elaborar um convite formal para os professores, obtendo o consentimento livre e esclarecido para participar da pesquisa. Durante o agendamento das entrevistas, também foram utilizados o Whatsapp e o e-mail.

Sobre o desenvolvimento do roteiro foi estimado uma média de 20 minutos de conversa e a condução foi dividida em 6 estágios, que serão descritos a seguir.

- **Perguntas Iniciais:** Nesse momento foi possível entender mais sobre a pessoa que estava sendo entrevistada. Foram perguntados aspectos como área de atuação e de estudo, experiências prévias sobre IHC e UX/UI.

- Exemplo: *"Qual é a sua área de estudo/trabalho?"*
- **Sobre a USINN:** Buscou entender a frequência do uso da ferramenta USINN Modeler e as dificuldades enfrentadas pelo usuário.
  - Exemplo: *"Já utilizou a USINN Modeller?"*
- **Dos recursos que precisam de tutorial:** Buscou entender quais recursos da plataforma precisam de mais urgência no desenvolvimento de ferramentas de tutorial.
  - Exemplo: *"Quais são os recursos da USINN Modeler que você sente mais dificuldades ao utilizar?"*
- **Recursos Ausentes:** Buscou entender quais recursos estão disponíveis em outras plataformas que poderiam ser implementados na USINN
  - Exemplo: *"Existem recursos de ajuda ou tutorial ou funcionalidades que você gostaria de ter na USINN Modeler, mas que atualmente não estão disponíveis?"*
- **Sobre ferramentas no geral:** Buscou entender quais ferramentas de diagramação utilizadas por outros usuários e as dificuldades enfrentadas nelas.
  - Exemplo: *"Além da USINN Modeller, quais outras ferramentas de diagramação /modelagem você já utilizou?"*
- **Extra:** Momento em que o entrevistado poderia indicar melhorias sobre a plataforma ou o trabalho que estava sendo desempenhado.

### 5.3 Condução das Entrevistas

Sommerville (2011) destaca duas características eficazes para um entrevistador que são: estar aberto a novas ideias, evitando ideias preconcebidas sobre os requisitos e estar disposto a ouvir os stakeholders. Mesmo que o stakeholder apresente requisitos-surpresa, eles estão dispostos a mudar de ideia sobre o sistema. Esse ponto é muito valioso e durante a entrevista deixá-lo expressar sua opinião mesmo que muitas vezes não seja algo que está diretamente ligado a recursos de tutorial o ajuda a entender o que pode ser melhorado no sistema e por vezes o levou a epifania. Em consonância não deixar que o entrevistado complete seu raciocínio pode privar o entrevistado de percepções futuras sobre o sistema.

O segundo comentário a qual Sommerville (2011) tece é que o entrevistado deverá participar de discussões com uma questão-trampolim, uma proposta requisito ou trabalhando em conjunto em um protótipo do sistema. Assim, semelhante ao que foi evidenciado no tópico anterior, poder dividir a entrevista em momentos interligados entre sessões com o convidado



possibilitou criar uma estrutura lógica de questionamento fornecendo uma melhor extração de informações.

Sobre as entrevistas foram conduzidas com cinco convidados ao todo e todas gravadas utilizando o *Google Meet*, de forma que com o consentimento e conhecimento dos entrevistados foi possível gerar os artefatos gerados após as entrevistas entre elas as gravações e anotações a qual poderão ser vistas nos apêndices A, B, C, D, E.

Após a conclusão das entrevistas foram realizadas as transcrições das entrevistas as quais foi utilizado a ferramenta Transkriptor que é uma ferramenta com o auxílio de inteligência artificial transcrever o áudio da gravação para um arquivo de texto. Os dados coletados durante as entrevistas podem ser acessados pelo link [https://drive.google.com/drive/folders/1bLVLGWf3Vp-JW2gPKuGWIqEPVYG-cRvB?usp=share\\_link](https://drive.google.com/drive/folders/1bLVLGWf3Vp-JW2gPKuGWIqEPVYG-cRvB?usp=share_link).

#### **5.4 Análise qualitativa dos dados**

Segundo Gibbs (2009), os dados qualitativos são essencialmente significativos, mas, mais do que isso, mostram grande diversidade. Eles não incluem contagens e medidas, mas sim praticamente qualquer forma de comunicação humana - escrita, auditiva ou visual; por comportamento, simbolismos ou artefatos culturais.

Ele também afirma que é importante registrar o rótulo ou nome do código usado para marcar e codificar a transcrição. Além disso, a definição do código é uma descrição da ideia analítica que ele refere e formas de garantir que a codificação seja confiável, ou seja, realizada de forma sistemática e constante. Ele ainda afirma sobre a forma indutiva de pesquisa a qual a amostragem de casos, contextos ou entrevistados é guiada pela necessidade de testar os limites de explicações que estão em desenvolvimento, as quais são constantemente baseadas nos dados que estão sendo analisados.

Os conceitos apresentados acima são de suma importância e serão base para o desenvolvimento da análise que será sucesso. Para a análise dos dados coletados, os participantes foram nomeados de P1 até P5 e também foi realizada uma análise da transcrição das entrevistas que podem ser acessadas pelo link compartilhado acima.

Gibbs (2009) afirma que a abordagem indutiva seria ler o texto, identificar um tema ou conteúdo que possa ser codificado e criar um novo código para ele, codificando o texto imediatamente. Assim, a partir das indagações acima, o desenvolvimento da pesquisa sucedeu com a codificação aberta, ou seja, foram criados códigos a partir da leitura dos textos em que foram criadas cores para representar a sua classificação.

## **5.5 Design e desenvolvimento**

A partir dos dados extraídos e das dificuldades identificadas nas etapas anteriores, foram modeladas e desenvolvidas as novas funcionalidades. Dessa forma, após discussão com a orientadora e o time da USINN Modeler entendeu-se a necessidade de melhorias em aspectos da página de criação e edição de diagramas. Assim, parte da solução foi levada até o design da equipe para que a equipe pudesse desenvolver a prototipação e em paralelo foi iniciado o processo de avaliação da ferramenta atual.

Durante a avaliação da plataforma foi identificado que a biblioteca mxgraph estava descontinuada e que vide a alta rotação de desenvolvedores no projeto o código atual para criar e editar diagramas era pouco dominado pela equipe. A partir disso, identificou-se a possibilidade de refatoração dessa funcionalidade e vide a complexidade foi necessário pesquisas e avaliações de outras bibliotecas para diagramação com javascript .

Esse processo contou com uma avaliação sobre aspectos do open-source. A comunidade ativa e o modelo de desenvolvimento contínuo são aspectos positivos fundamentais do software de código aberto. A colaboração em comunidades open source permite que desenvolvedores de todo o mundo contribuam para o aprimoramento constante de bibliotecas e ferramentas, resultando em software de alta qualidade e atualizado (RAYMOND, 2001). Assim, visando o benefício do projeto foram levados os seguintes aspectos que são demonstrados na figura abaixo e serão posteriormente explicados.

Figura - 5 Comparação entre mxgraph e reactflow

comparações	mxgraph	reactflow
forks	2.1k	1.6k
star	6.8k	24.7k
usado por	2.1k	5.7k
documentação robusta	✗	✓
Utilizar javascript/html/css para criar componentes	✗	✓

Fonte: Elaborado pelo autor (2024).

A figura acima representa a *mxgraph* a qual é utilizada atualmente pela USINN Modeler para o desenvolvimento de diagramas e a *reactflow* que seria a candidata a substituí-la. Ademais, também é importante ressaltar o descontinuação da ferramenta *mxgraph*, ou seja, não é mais mantida pela comunidade. É válido ressaltar que as informações acima foram extraídas da ferramenta de versionamento Github.

- **Forks:** É uma medida importante pois avalia o impacto da ferramenta para a comunidade, demonstrando um certo grau de aceitação por parte dela a qual julga importante e a partir dele é possível criar novas soluções que sirvam ao nicho do desenvolvedor.
- **Star:** Seria a quantidade de estrelas a qual é atribuído um projeto, a forma para a comunidade conseguir avaliar o projeto.
- **Usado por:** Quantidade de pessoas que são dependentes do projeto em questão em seus repositórios, demonstrando uma comunidade mais ativa, diminuindo as chances da ferramenta de cair em desuso e descontinuada.

- **Documentação robusta:** Para avaliar esse processo foi levado em consideração, principalmente, a quantidade de exemplos entregues pela documentação de forma a exemplificar o seu uso.
- **Utilizar javascript/html/css para criar componentes:** Esse foi o aspecto mais importante durante a escolha e para definir qual das ferramentas disponíveis seriam utilizadas pois é de senso comum para os desenvolvedores do projeto o uso dessas linguagens e é de suma importância garantir que a ferramenta utilize tecnologias bem conhecidas a fim de garantir a longevidade do projeto. Como pode ser observado abaixo.

Figura - 6 Trecho de código mxgraph

A screenshot of a code editor window with a dark background and light-colored text. The code is XML for mxgraph, defining a shape named 'presentationUnit'. The code includes a background element with a rounded rectangle and a foreground element with a fillstroke. The code is as follows:

```
<shape name="presentationUnit" strokewidth="inherit" h="300" w="400">
  <background>
    <roundrect h="300" w="400" x="0" y="0" arsize="5"/>
  </background>
  <foreground>
    <fillstroke/>
  </foreground>
</shape>
```

Fonte: Elaborado pelo autor (2024).

Figura - 7 Trecho de código refatorado

```
<div
  style={{
    width: isMinimized ? 230 : 500,
    height: isMinimized ? 50 : 300,
    backgroundColor: 'rgba(128, 128, 128, 0.2)',
    border: '2px solid #999',
    borderRadius: '10px',
    position: 'relative',
  }}
>
  <div
    style={{
      display: 'flex',
      alignItems: 'center',
      justifyContent: 'space-between',
    }}
  >
    <>
      {isEditing ? (
        <input
          type="text"
          value={text}
          onChange={handleChange}
          onBlur={handleBlur}
          autoFocus
          style={{ position: 'absolute', top: 10, marginLeft: 40 }}
          onKeyDown={handleKeyDown}
        />
      ) : (
        <span
          style={{ position: 'absolute', top: 10, marginLeft: 40 }}
          onClick={handleClick}
        >
          {text}
        </span>
      )}
    </>
    <Button
      onClick={toggleMinimize}
      style={{
        minWidth: 'auto', // Para ajustar o tamanho do botão ao ícone
      }}
    >
      {isMinimized ? <AddIcon /> : <RemoveIcon />}
    </Button>
  </div>
</div>
```

Fonte: Elaborado pelo autor (2024).

As imagens 6 e 7 acima seriam uma breve representação do código fornecido com mxgraph e reactflow para o desenvolvimento da mesma notação, a Unidade de Apresentação. Como pode ser observado na figura 6, o código em mxgraph é menos verboso porém para a criação foi necessário utilizar uma notação própria da ferramenta e em contrapartida a figura 7 representa o código já refatorado e apesar de um código mais verboso o conhecimento prévio dos desenvolvedores em HTML, CSS e Javascript torna a escolha dessa ferramenta mais fácil para o desenvolvimento das manutenções futuras.

Além disso, outro passo importante para a implementação foi o entendimento da notação, ou seja, compreender quais conectores podem ser emitidos e recebidos pelos elementos. Para essa finalidade foi fornecido pela professora orientadora uma tabela semelhante à figura abaixo demonstrando essas regras.

Figura - 8 Regras USINN

Categoria	Elementos	Conectores Válidos	
		Emitir	Recebe
Navegação	Unidade de Apresentação	Navegação (Apenas na versão minimizada)	Navegação (Apenas na versão minimizada)
Navegação	Ponto de Abertura	Navegação	-
Navegação	Ponto de Fechamento	-	Navegação
Interação	Ação do Usuário	Transição	Transição; Feedback do Sistema; Query de Dados;
Interação	Processo do Sistema	Feedback do Sistema; Transição de Cancelamento; Dados;	Transição do Usuário; Query de Dados;
Usabilidade	Unidade de Apresentação (Sempre Acessível)	Navegação	Navegação
Usabilidade	Ação do Usuário (Obrigatória)	Transição	Transição; Feedback do Sistema; Query de Dados;
Usabilidade	Alerta de Notificação ou Confirmação	Transição; Feedback do Sistema;	Transição; Feedback do Sistema;
Usabilidade	Coleção de Dados e Query	Query de Dados	Query de Dados
Usabilidade	Processo do Sistema com Indicador de Progresso	Feedback do Sistema; Transição; Query de Dados;	Transição; Query de Dados;

Fonte: Elaborado pelo autor (2024).

Como foi explicado em tópicos anteriores a definição dos elementos também foi necessário compreender a sintaxe da USINN e a tabela com as definições acima foram de suma importância para o desenvolvimento de um projeto correto.

Sobre a prototipação, durante a análise qualitativa desempenhada foram extraídas algumas falas como "Personalização, acho que de cores e tal. O estilo dos botões. Aparência, acho que é mais aparência.", assim tais afirmações demonstram por parte do usuário uma necessidade de melhorias em aspectos visuais da plataforma e, portanto, visando tais melhorias e tornar mais intuitivo a escolha de notação foram prototipados mudanças do menu. Como pode ser demonstrado na figura abaixo.

Figura - 9 Protótipo desenvolvido no Figma



Fonte: Elaborado pelo autor (2024).

Por conseguinte, foi iniciado o processo de desenvolvimento a qual foi escolhido a biblioteca reactflow para o desenvolvimento dos elementos para diagramação e para estilização a biblioteca Material-UI vide as inúmeras estilizações e funcionalidades nativas acompanhadas a qual serão demonstradas no capítulo posterior de resultados. E por fim, os testes realizados pela equipe de qualidade a qual puderam prover informações com base em documentações anteriores se a ferramenta desenvolvida até então estava alinhada com as necessidades.

## 6 RESULTADOS

A presente seção busca apresentar os resultados recolhidos a partir da análise qualitativa e o que foi desenvolvido e melhorado demonstrando algumas soluções para os problemas encontrados em testes anteriores.

### 6.1 Entrevistas e análise qualitativa

Informações extraídas a partir das entrevistas como os códigos com suas respectivas explicações e exemplos da classificação. Segue abaixo a tabela.

Tabela 1 - Resultado da análise qualitativa

<b>CÓDIGO</b>	<b>DESCRIÇÃO</b>
<b>ATIVIDADES DESEMPENHANDO ATUALMENTE OU A QUAL ATUOU</b>	Esse código representa as competências acadêmicas desempenhadas pelo entrevistado.
<b>NÍVEL DE CONHECIMENTO EM UX/UI OU IHC</b>	Esse código representa o nível de conhecimento do entrevistado em IHC e UX/UI. É válido ressaltar que não foi estabelecido um padrão baseado em tempo de estudo, de forma que ficou aberto para o usuário definir o seu nível de conhecimento na área.
<b>DIFICULDADES PARA ENTENDER SOBRE UX/UI</b>	Esse código representa as principais dificuldades dos entrevistados durante o aprendizado de UX/UI.
<b>FERRAMENTAS PARA ESTUDO DE UX/UI</b>	Esse código representa as principais ferramentas utilizadas pelos usuários de UX/UI para o aprendizado.
<b>EXEMPLO DE FERRAMENTA DE DIAGRAMAÇÃO</b>	Esse código representa outras ferramentas de diagramação utilizadas pelos entrevistados.
<b>FREQUÊNCIA DO USO DA USINN MODELER</b>	Esse código representa a frequência que os entrevistados utilizaram a USINN



	Modeler.
MELHORIAS PARA A PLATAFORMA	Esse código representa as melhorias apontadas pelos usuários para fazer na plataforma web.
EXEMPLOS DE IMPLEMENTAÇÃO UTILIZADOS POR OUTRAS PLATAFORMAS	Esse código representa funcionalidades presentes em outras plataformas web que poderiam ser implementadas de forma semelhante apesar de não serem plataformas de diagramação necessariamente.
QUESTIONAMENTO SOBRE USABILIDADE	Esse código representa os questionamentos feitos durante a entrevista por parte dos usuários durante a entrevista sobre a plataforma web USINN Modeler.
DIFICULDADES PARA UTILIZAR A PLATAFORMA	Esse código representa as dificuldades enfrentadas pelo usuário durante o uso da plataforma na entrevista.
DIFICULDADES EM OUTRAS FERRAMENTAS	Esse código representa as dificuldades enfrentadas pelo entrevistado em outras plataformas.

Fonte: Elaborado pelo autor (2023).

Para uma melhor visualização do resultado da análise qualitativa, a partir dos códigos acima foram extraídas parte dos trechos das entrevistas das quais os códigos foram marcados. Os códigos criados tiveram como objetivo destacar o nível de conhecimento do ecossistema a qual a ferramenta está inserida, entender as dificuldades enfrentadas pelo usuário não só utilizando a ferramenta em estudo, mas outras de maneira que esse fator possa ser levado em consideração durante o desenvolvimento das funcionalidades de ajuda. Abaixo segue algumas das respostas extraídas das entrevistas.

- **NÍVEL DE CONHECIMENTO EM UX/UI OU IHC:**
  - **P1:** *"...me considero iniciante"*
  - **P2:** *"Eu estou no nível mais intermediário"*
- **ATIVIDADES DESEMPENHANDO ATUALMENTE OU A QUAL ATUOU:**

- **P1:** *"Eu faço faculdade lá na UFC, no curso de Ciência da Computação. Tenho experiência justamente na USINN como QA, nessa parte de testes e garantia da qualidade"*
  - **P2:** *"...professor aqui na UNIPAMPA"*
- **DIFICULDADES PARA ENTENDER SOBRE UX/UI:**
    - **P1:** *"aplicar a teoria na prática."*
    - **P2:** *"Então, eu acho que fazer essa observação completa da experiência do usuário é a principal dificuldade. "*
- **FERRAMENTAS PARA ESTUDO DE UX/UI:**
    - **P1:** *"Lucy Chart, o BRModelo, Figma também"*
    - **P2:** *"Então, mas quando eu quero ter uma visão geral de técnicas de UX, olhando para fins de estudos, eu saio catando mesmo na própria escola ou eu abro o All About UX, que é um site bem interessante, que agora ele mudou de nome. Inclusive, eu posso até... All About UX"*
- **EXEMPLO DE FERRAMENTA DE DIAGRAMAÇÃO:**
    - **P1:** *"tipo uma vez por semana, ou até duas, por conta do da minha escrita também do TCC"*
    - **P3:** *"O BPMN Modeling."*
- **FREQÜÊNCIA DO USO DA USINN MODELER:**
    - **P1:** *"tipo uma vez por semana, ou até duas, por conta do da minha escrita também do TCC"*
    - **P3:** *"colocaria de duas a três vezes por mês."*
- **MELHORIAS PARA A PLATAFORMA:**
    - **P1:** *"... exportar algum diagrama que realizou lá, em formato de imagem, é tudo embaçado, literalmente embaçada a imagem."*
    - **P3:** *"Personalização, acho que de cores e tal. O estilo dos botões. Aparência, acho que é mais aparência."*
- **EXEMPLOS DE IMPLEMENTAÇÃO UTILIZADOS POR OUTRAS PLATAFORMAS QUE NÃO SÃO DE DIAGRAMAÇÃO NECESSARIAMENTE:**
    - **P3:** *"... um tutorial apresentando os elementos da anotação, como utilizar, categorizar bem certinho. Um vídeo. E um PDF resumido, para quem não gosta de vídeo. Uma apresentação resumida em documento."*
    - **P4:** *"...mas eu acho que falta tipo um manual de instruções para dizer justamente o que eu falei. De tipo, dizer o que cada determinada coisa faz. E também como a pessoa poderia fazer cada coisa na USINN, como acessar os diagramas, criar um diagrama. Como se fosse um tutorial básico mesmo, mas num pequeno manual."*
- **QUESTIONAMENTO SOBRE USABILIDADE:**
    - **P1:** *"Só olhando para essa tela, por que aqui dentro do meu perfil tem um excluir perfil? Ai eu excluí meu perfil? Sério que eu excluí? Tu*

*entendeu o ponto que eu mencionei?"*

- **DIFICULDADES PARA UTILIZAR A PLATAFORMA:**

- **P1:** *"Mas às vezes para quem está realizando o diagrama, isso não fica tão claro. De qual unidade tem que ser tem que ser colocada para realizar o diagrama. Se é a unidade de apresentação ou unidade de apresentação que é sempre acessível."*
- **P3:** *"Às vezes, a questão de como usar a anotação, sim. O fato da ferramenta não é o mais dificultoso. E como você está usando certa anotação"*

- **DIFICULDADES EM OUTRAS FERRAMENTAS:**

- **P1:** *"Uma das maiores dificuldades foi de, tipo, construir meio que relações certinhas porque ele meio que não avisava tanto. Eu tive um pouco de dificuldade assim. Eu não sei nem se eles meio que repararam esse erro, porque está com o tempo que eu utilizei. E não acho que no Figma meio que a gente exporta os elementos. "*
- **P5:** *"Será que estou fazendo certo? Alguns diagramas são assim, tipo, você tem que ter uma experiência naquele diagrama. Já outros não."*

## 6.2 Refatoração da USINN Modeler

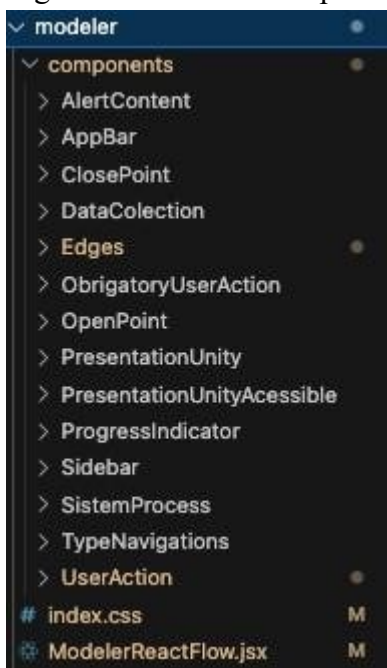
Nesse tópico serão apresentadas as escolhas técnicas e exemplos da tecnologia utilizada e como foi implementada. Além da estrutura de pastas, exploração das componentes desenvolvidas, novas implementações, aplicação das regras de negócio e escolhas técnicas.

Também é importante clarificar que a principal engrenagem do desenvolvimento é a biblioteca reactflow, que conta com o conceito semelhante ao conceito de grafos, envolvendo nós e arestas. Os nós seriam os elementos que recebem ou emitem conexões e as arestas são os diversos tipos de conexões que serão melhor explorados posteriormente.

- **Organização de diretórios:**

Na raiz do diretório é possível encontrar o arquivo "ModelerReactFlow" o qual vai conter a "engrenagem" com as regras principais. Neste arquivo será possível definir regras como "o que fazer ao deletar um elemento do diagrama?", "quais elementos podem emitir e receber conexão?". Ademais, dentro dela estarão todos os recursos auxiliares necessários para que possa funcionar corretamente como o diretório Componentes que vai conter a estrutura dos elementos da notação, como pode ser observado na Figura 10.

Figura - 10 Estrutura de pastas da refatoração



Fonte: Elaborado pelo autor (2024).

É importante salientar que foi utilizado as técnicas de clean code para nomeação dos diretórios, assim dentro de componentes terá a representação da notação, incluindo os nós e arestas, ou seja, "AlertContent" equivale a Alerta de Notificação, "ClosePoint" a Ponto de fechamento, "OpenPoint" a Ponto de abertura, "DataCollection" a Coleção de Dados e Query, "ObrigatoryUserAction" a Ação obrigatória do usuário, "PresentationUnity" a Unidade de Apresentação, "PresentationUnityAcessible" a Unidade de Apresentação Acessível, "ProgressIndicator" a Processo do Sistema com Indicador de Progresso, "SistemProcess" a Processo do sistema e "UserAction" a Ação do Usuário.

A "Sidebar" seria o menu de seleção da notação a qual foi implementado com base no protótipo apresentado anteriormente e a "AppBar" contendo a interface visual para salvar, baixar o diagrama e modificar o nome do diagrama.

Por fim a "*TypeNavigations*" que em tradução seriam os "tipos de navegação", a qual irá conter uma regra de negócio e escolha técnica para o projeto, a validação visual de quais

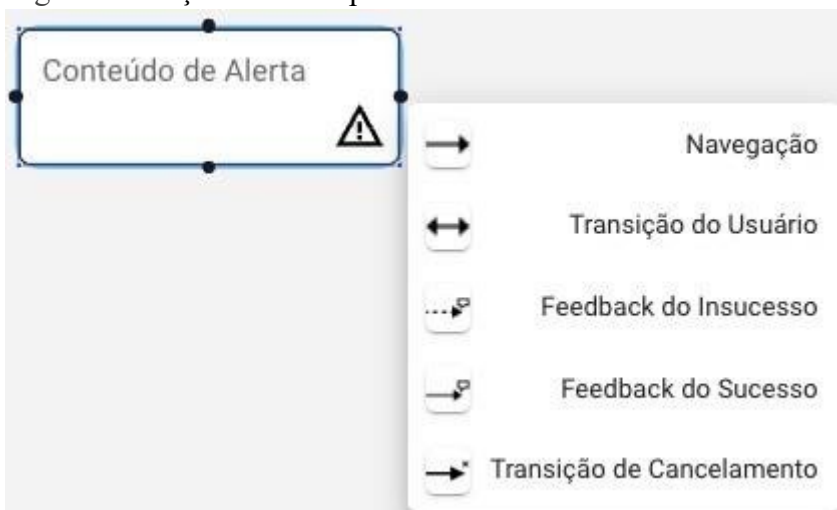
campos o usuário poderá selecionar para conectar. Anteriormente, o usuário tinha a escolha de selecionar conectar com quaisquer elementos do diagrama e para sinalizar que aquela ação não estava correta era emitido um alerta para ele evidenciando o erro. Nessa versão optou-se por uma nova abordagem a qual impede o usuário de cometer o equívoco.

Figura - 11 Ação ao clicar para selecionar conexão



Fonte: Elaborado pelo autor (2024).

Figura - 12 Ação ao clicar para selecionar conexão



Fonte: Elaborado pelo autor (2024).

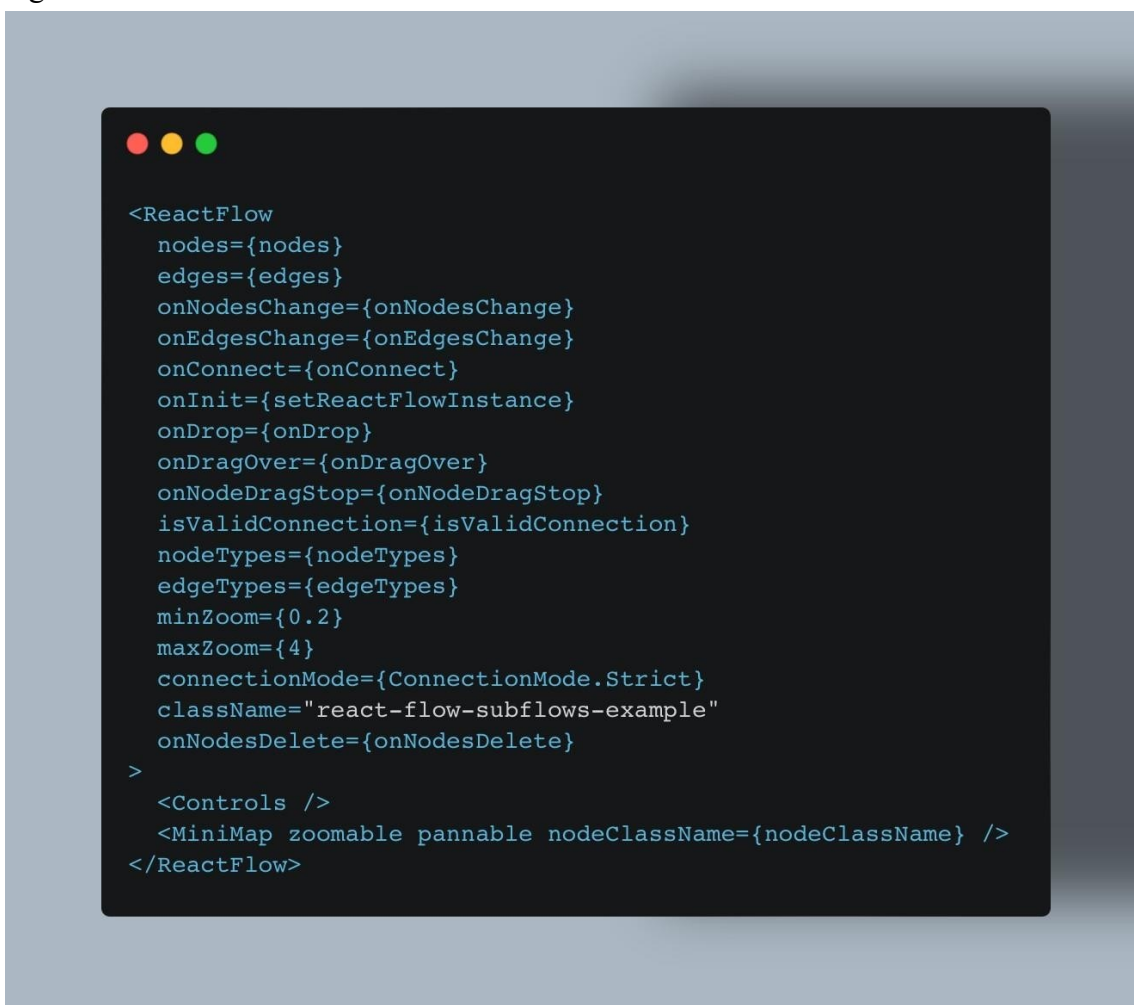
Assim, antes do usuário tentar fazer qualquer ligação entre os elementos ele deve selecionar o tipo de conexão que ele pretende realizar e para cada elemento da notação será demonstrado o conjunto adequado de conexões que ele pode realizar. É válido ressaltar como essa função pode agregar ao usuário, pois como ser visto nas entrevistas do tópico 6.1 aos participantes quando apontado sobre dificuldades da plataforma o participante P1, "*Mas às vezes para quem está realizando o diagrama, isso não fica tão claro.*" e o P3, "*Será que estou*

*fazendo certo? Alguns diagramas são assim, tipo, você tem que ter uma experiência naquele diagrama. Já outros não.*", logo visou-se melhorar a experiência do usuário impedindo que ele pudesse cometer o erro.

- **Estrutura do reactflow:**

Essa seção tem como objetivo prover uma melhor explicação sobre o como a ferramenta foi adaptada para as necessidades da USINN. Para um melhor entendimento segue a figura abaixo.

Figura - 13 Estrutura base reactflow



Fonte: Elaborado pelo autor (2024).

Como foi citado anteriormente o reactflow tem como base os princípios de grafos, assim para saber quais elementos ele deve mostrar para o usuário ele espera receber uma lista de nós e arestas e para que possamos visualizar o que tipo de lista que ele está criando segue exemplificação com o que contém em cada item na lista de nós e arestas, respectivamente.

Figura - 14 Estrutura de dados de um nó

```
{
  "id": "id-2e459b79-ea71-4b7f-9fa8-a549c5a4ac07",
  "type": "user-action",
  "position": {
    "x": 230.7329559326172,
    "y": 158.01136779785156
  },
  "data": {
    "label": "user-action-node",
    "name": ""
  },
  "dragging": false,
  "width": 163,
  "height": 50,
  "selected": false,
  "positionAbsolute": {
    "x": 230.7329559326172,
    "y": 158.01136779785156
  }
}
```

Fonte: Elaborado pelo autor (2024).

Figura - 15 Estrutura de dados de uma aresta

```
{
  "source": "id-b1907e70-c4e1-430b-8d7c-da132f6546ff",
  "sourceHandle": null,
  "target": "id-2e459b79-ea71-4b7f-9fa8-a549c5a4ac07",
  "targetHandle": "user-action-target-left",
  "type": "navigation",
  "id": "reactflow__edge-id-b1907e70-c4e1-430b-8d7c-da132f6546ff-id-2e459b79-ea71-4b7f-9fa8-a549c5a4ac07user-action-target-left"
}
```

Fonte: Elaborado pelo autor (2024).

Para os exemplos acima a primeira figura representa um nó, ou seja, ele possui um identificador único, o tipo de nó que deve ser mostrado ao usuário de acordo com o que ele selecionou no menu de elementos, sua posição na tela, se ele deve possuir informação ao campo de texto, altura, largura, se ele foi selecionado e se ele eles estão sendo arrastado.

Figura - 16 Exemplo da notação após a refatoração



Fonte: Elaborado pelo autor (2024).

Esse seria o resultado, de forma que foi omitido o ponto de abertura, visto que teria uma configuração semelhante ao de ação do usuário a diferir o tipo de nó a qual está sendo inserido a lista, o *'open-point'*.

Ademais conta com a função *'onNodesChanges'* e *'onEdgesChange'* que serão emitidas sempre que houver quaisquer edições ao diagrama seja adição, deleção ou mudança de elemento. A função *'onConnect'* é emitida sempre que um elemento é conectado a outro com sucesso pois será nela que a nova aresta será inserida na lista. A função *'onDrop'* e *'onDragStart'* atuam em conjunto, a *'onDragStart'* ela está presente na *'Sidebar'* e sempre que um elemento do menu é arrastado ele irá emitir esse evento com o tipo para que possa ser referência a função *'onDrop'* e possa identificar qual o tipo de elemento está sendo inserido na tela. A função *'onNodeDragStop'* é de suma importância principalmente para a notação de unidade de apresentação e unidade de apresentação acessível, pois o reactflow apresenta o conceito de subflow que seria a forma de associar que um nó está aninhado a outro e portanto essa funcionalidade irá verificar se a posição do nó que está sendo arrastado não é do tipo unidade de apresentação e unidade de apresentação acessível e se está sobrepondo ela, assim se tornando um um nó filho daquela notação. Para que possa fazer o desagrupamento foi implementado uma função a qual verifica o nó selecionado e ao clicar com o botão direito do mouse aparecerá um menu com a opção "desagrupar". A função *'isValidConnection'* funciona para validar se um elemento pode se conectar a outro e será melhor explicada nos próximos tópicos. E por fim, a função *'onNodesDelete'*, desenvolvida para lidar com a deleção de elementos, foi criada para garantir que elementos de subflow não sejam deletados caso o seu nó pai seja deletado.



- **Nós e arestas customizados:**

O reactflow irá apresentar nós e arestas nativos, porém além disso ele permite ao desenvolvedor conseguir customizá-los e para o projeto foi de suma importância para conseguir adaptar a interface das notações.

- **Validação de conexões:**

A validação das notações foi realizada de acordo com o documento previamente apresentado para melhorar exemplificar a lógica, segue trecho de código.

Figura - 17 Regra contendo validação entre conexões



```

const nodeSource = nodes.find(nd => nd.id === connection.source);
const nodeTarget = nodes.find(nd => nd.id === connection.target);

if(!nodeSource || !nodeTarget) return false;
if(currentEdge.length === 0) return false;

let sourceClause = false;
let targetClause = false;

if(nodeSource.type === "open-point") {
  sourceClause = currentEdge === "navigation";
}

if(nodeSource.type === "user-action") {
  sourceClause =
    currentEdge === "transition" ||
    currentEdge === "cancel-transition" ||
    currentEdge === "navigation"
}

if(nodeTarget.type === "user-action") {
  targetClause =
    currentEdge === "transition" ||
    currentEdge === "sucess-feedback" ||
    currentEdge === "unsucess-feedback" ||
    currentEdge === "query-data" ||
    currentEdge === "cancel-transition" ||
    currentEdge === "navigation"
}

```

Fonte: Elaborado pelo autor (2024).

Assim essencialmente esse trecho de código representa que inicialmente será buscado quem é a fonte da conexão e que está recebendo a conexão ("*nodeSource*" e "*nodeTarget*"), em seguida duas variáveis são iniciadas como falsos ('*sourceClause*' e '*targetClause*'). Por fim, é comparado o tipo de elemento que está recebendo e emitindo, caso ambos não atendam as validações e não se tornem verdadeiros ao longo da validação a conexão é inválida e, portanto, impedindo que seja criado.

Logo, para o exemplo que foi mostrado na Figura 17 essa função de validação recebe *connection* como parâmetro e em seguida é procurado na lista de nós qual o tipo que está tentando emitir (*nodeSource*) e receber (*nodeTarget*). Em seguida é feito validações duas validações a primeira é para impedir que tente fazer conexão entre um tipo não seja um dos customizados que foram criados e o segundo para impedir de tentar

fazer conexão se não for uma aresta do tipo customizada.

Em seguida são iniciadas duas variáveis como falsas, representando se quem está emitindo é válido e quem está recebendo também. Portanto, no caso da Figura 15, foi verificado que o tipo de nó que está emitindo é *open-point* e quem está recebendo é *user-action* e o tipo de navegação (*currentEdge*) que está tentando ser feita é do tipo *navigation*. Ao final se ambas forem verdadeiras a conexão é realizada e a aresta é inserida na lista de *edges*.

- **Integração com backend:**

Nessa seção é válido ressaltar que para fazer a integração com o backend da aplicação foram necessários realizar a mudança de 2 rotas, a de criação e edição pois previamente para que conseguisse salvar informações sobre o diagrama era necessário salvar a notação do *mxgraph* porém na atual será necessário salvar listas de arestas e nós, contendo as informações referentes às adaptações da notação. Além disso também foi necessário inserir a coluna *data* a tabela *diagram* do projeto

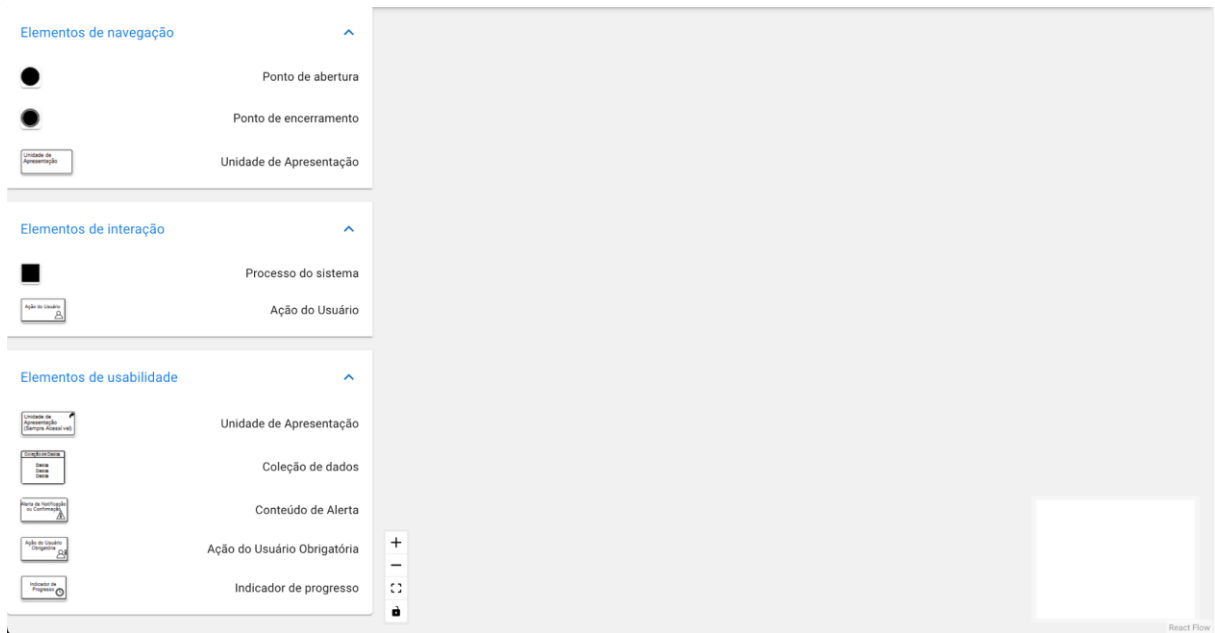
### 6.3 Avaliações preliminares

Os resultados preliminares indicam que a refatoração dos principais recursos da USINN Modeler trouxe avanços significativos em termos de clareza de código e manutenção. A utilização de práticas de código limpo e a escolha da biblioteca React Flow, que facilita a criação de diagramas interativos com maior controle sobre os elementos, possibilitaram uma organização modular e simplificada do código. Com a nova estrutura de diretórios e a adoção de uma arquitetura mais clara, foi possível reduzir a duplicidade de código e melhorar a legibilidade, facilitando o entendimento e a modificação por parte de novos desenvolvedores.

A adoção de uma nova solução com ampla documentação e suporte da comunidade permitiu a criação de funcionalidades robustas e escaláveis. Essa mudança garante que o projeto seja mais sustentável a longo prazo, permitindo futuras expansões e manutenções com menos esforço e maior confiabilidade, beneficiando tanto a experiência do desenvolvedor quanto a evolução contínua da ferramenta.

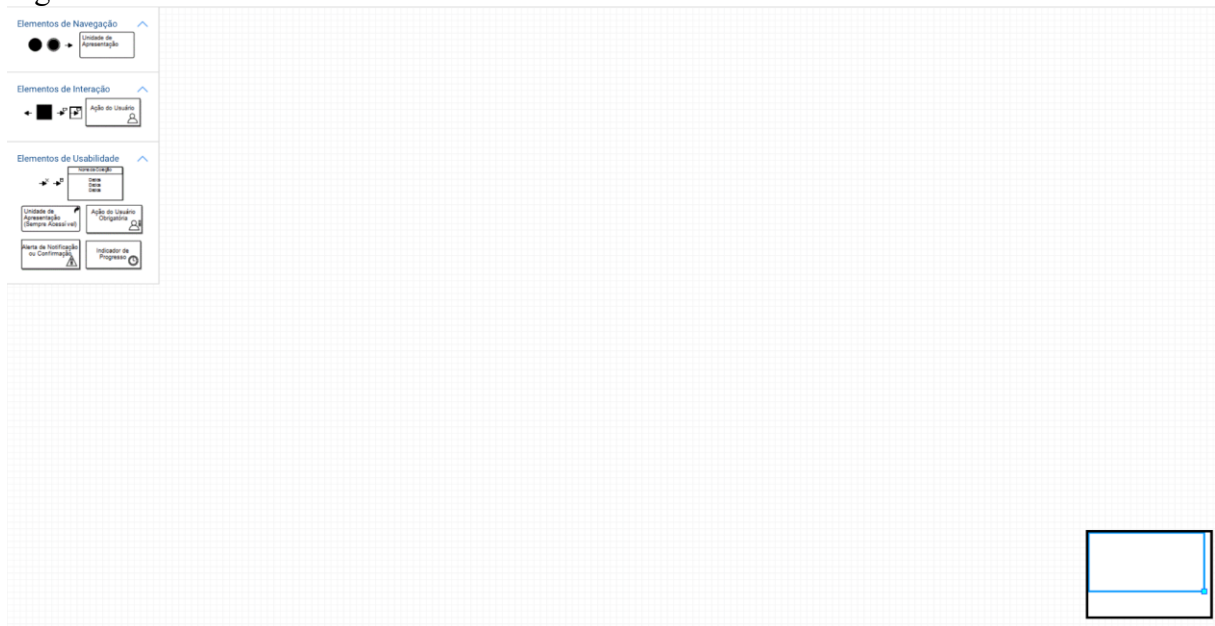
Pode-se demonstrar também algumas melhorias relacionadas a interface de apresentação das notações como foi prototipado e demonstrado nas seções anteriores.

Figura - 18 Menu refatorado



Fonte: Elaborado pelo autor (2024).

Figura - 19 Menu atual

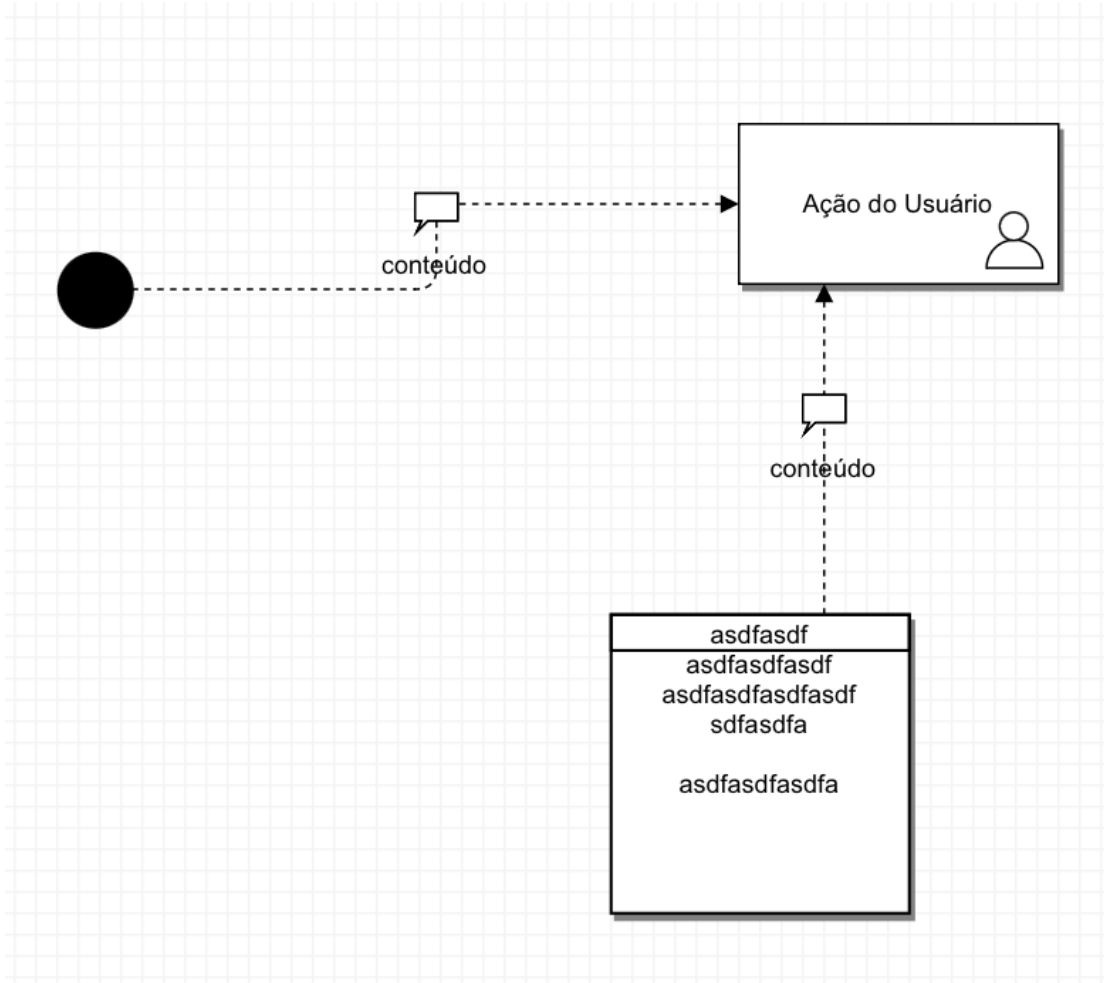


Fonte: Plataforma USINN Modeler (2024).

Como é ilustrado na Figura 18 é possível perceber um acompanhamento com o nome da notação com o objetivo de entregar mais informações ao usuário e como foi comentado em uma das entrevistas *"Personalização, acho que de cores e tal. O estilo dos botões. Aparência, acho que é mais aparência."*. Assim, a fim de atender a um dos requisitos indicados pelo usuário foi implementado essa interface em contrapartida a interface da Figura 19.

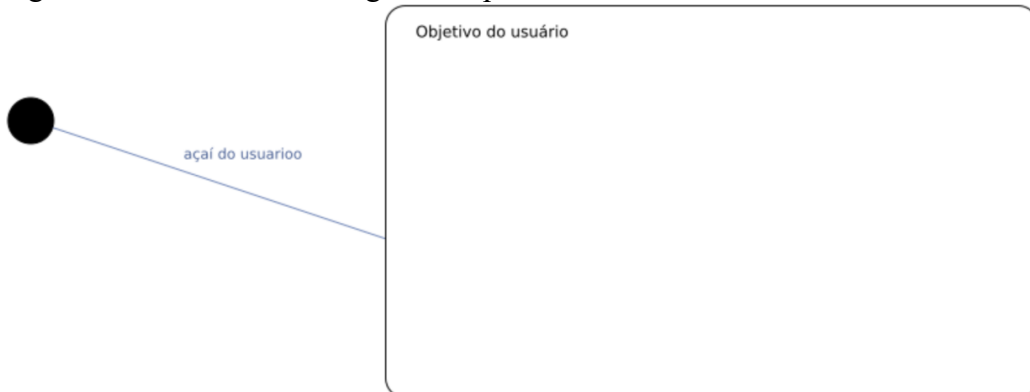
Outra solução implementada foi para exportar diagramas. Essa funcionalidade consiste em fazer o download de acordo com os formatos pdf, jpen e png a partir do que foi gerado com a notação na ferramenta. Assim espera-se que ao desenvolver um diagrama seja possível visualizar as informações completas.

Figura - 20 Diagrama teste para exporta



Fonte: Plataforma USINN Modeler (2024).

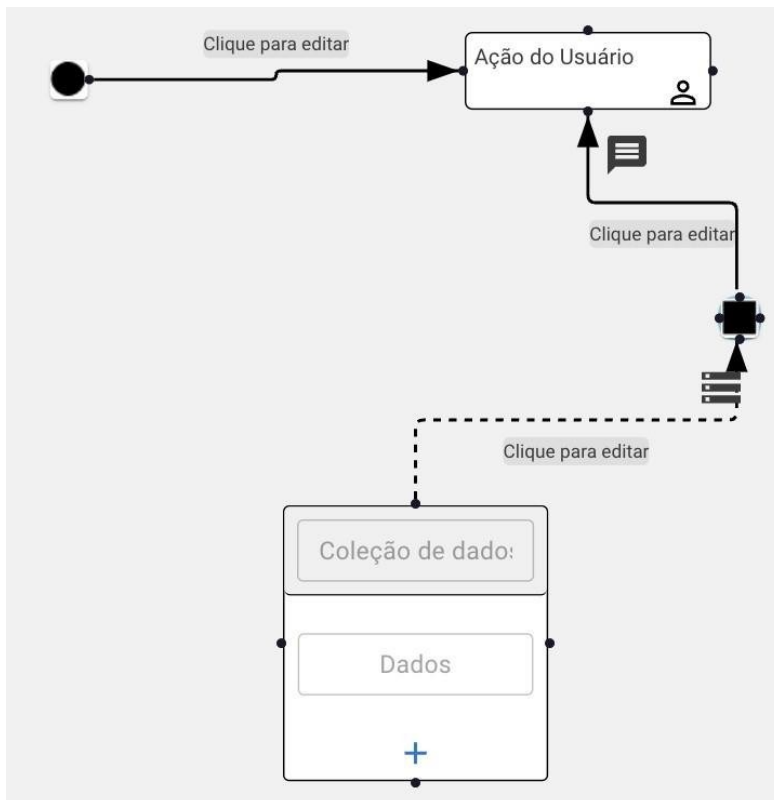
Figura - 21 Resultado do diagrama exportado



Fonte: Elaborado pelo autor (2024).

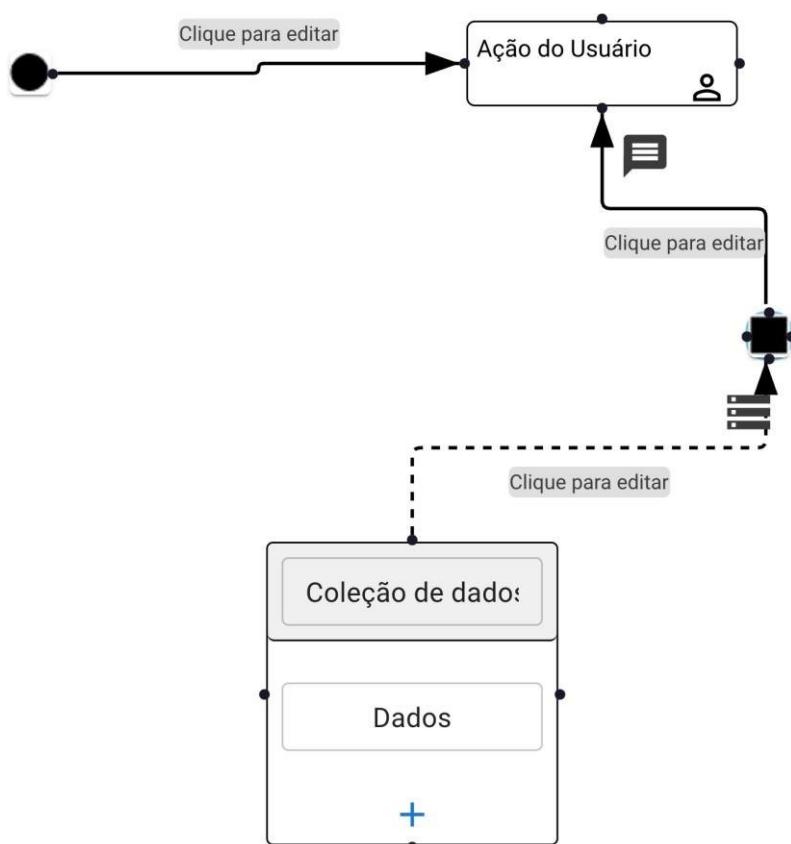
Para adquirir essas imagens foi seguido o seguinte fluxo, as componentes foram inseridas na tela, as conexões foram feitas e em seguida foi clicado para exportar e selecionado PNG. Assim, para o diagrama gerado na figura 20 foi resultado da exportação o diagrama da figura 21, demonstrando um bug por parte da plataforma. Ainda é válido agregar que essa funcionalidade pode ser considerada de suma importância pois também foi mencionada na entrevista, "... exportar algum diagrama que realizou lá, em formato de imagem, é tudo embaçado, literalmente embaçada a imagem."

Figura - 22 Diagrama de teste para exportar.



Fonte: Elaborado pelo autor (2024).

Figura - 23 Resultado do diagrama exportado do diagrama refatorado.

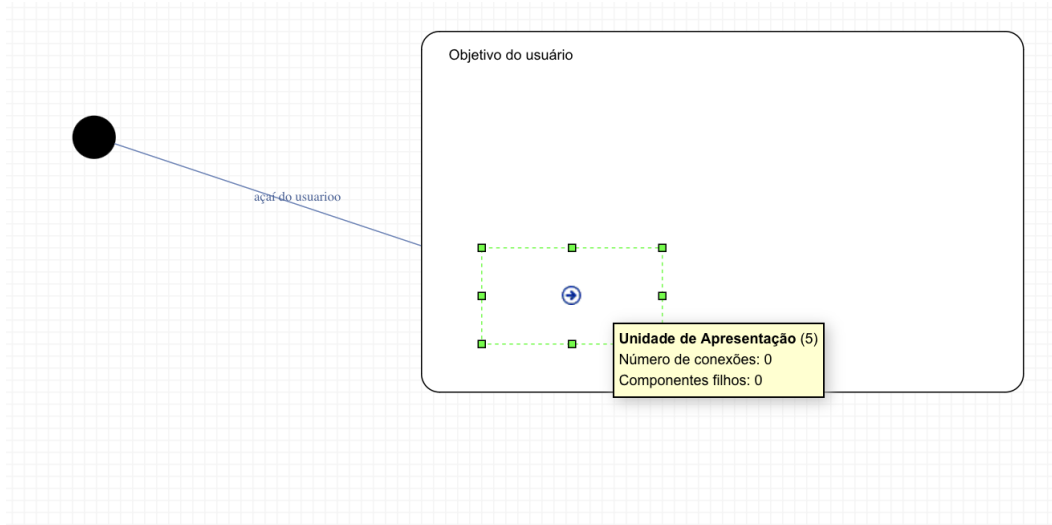


Fonte: Elaborado pelo autor (2024).

Para o desenvolvimento desse diagrama foram realizados os mesmos passos na plataforma com a refatoração, de forma que a figura 22 representa um print antes de realizar a exportação. Também deve ser relatado que diferente da versão em produção que aceita selecionar diversos tipos para exportar como pdf, jpeg e png. Nessa versão ao selecionar para exportar já é gerado em png. Assim o resultado é fiel ao diagramado. Outro ponto interessante sobre esse diagrama é o uso das validações pois seguindo o documento que foi explicado anteriormente coleção de dados deve emitir apenas conectores do tipo query de dados e não feedback, como foi demonstrado na figura 20.

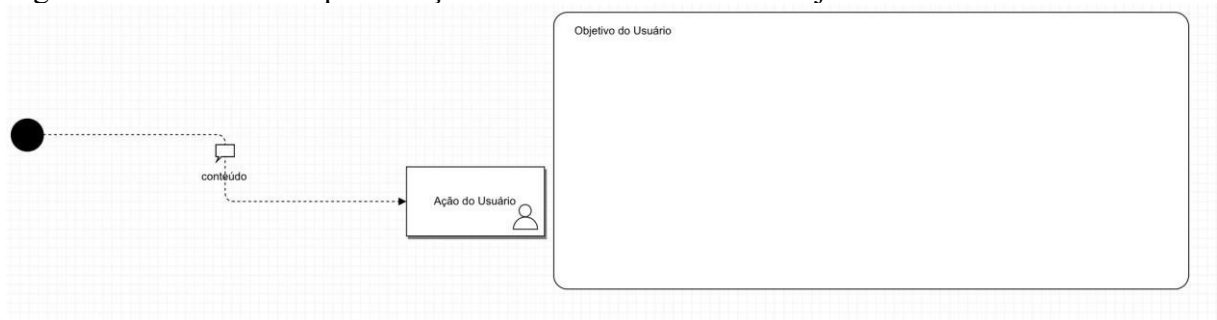
Outro aspecto interessante a qual foi corrigido nesta versão refatorada, foi a relação entre notações simples e a unidade de apresentação e unidade de apresentação acessível. Atualmente, em algumas situações ao inserir um elemento dentro do tipo unidade de apresentação ao arrastá-la por vezes o elemento se mantém no lugar e a unidade é arrastada. Ademais, também existem situações que a unidade de apresentação sobrepõe a notação simples, como será demonstrado no exemplo abaixo.

Figura - 24 Unidade de apresentação sobrepondo elemento.



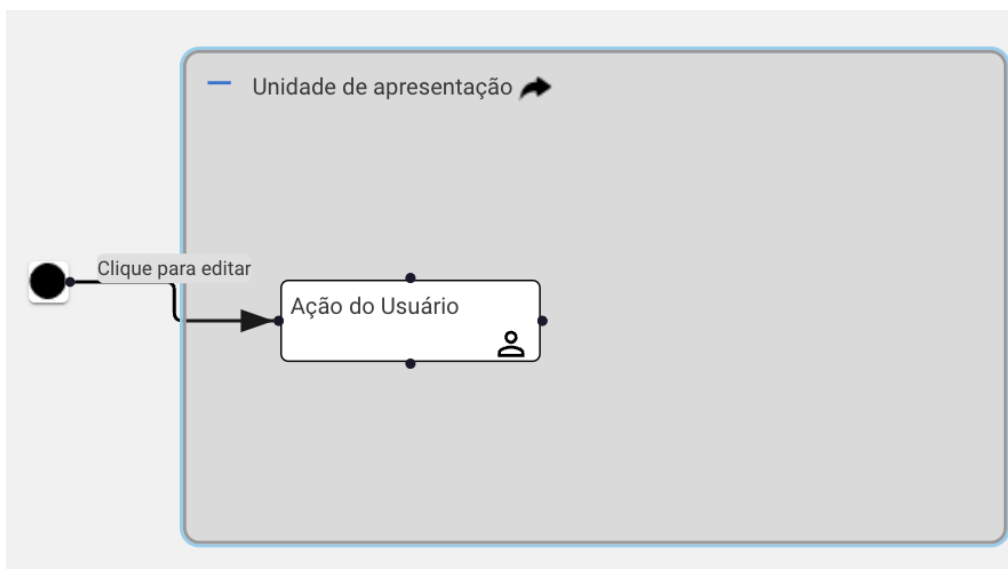
Fonte: Elaborado pelo autor (2024).

Figura - 25 Unidade de apresentação não arrastando elementos juntos.



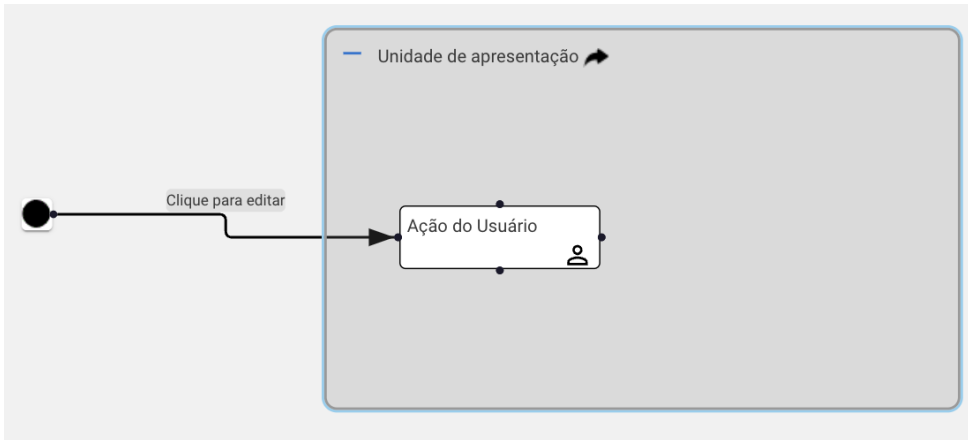
Fonte: Elaborado pelo autor (2024).

Figura - 26 Unidade de apresentação refatorada antes de ser arrastada.



Fonte: Elaborado pelo autor (2024).

Figura - 27 Unidade de apresentação refatorada após ser arrastada.



Fonte: Elaborado pelo autor (2024).

Como pode ser evidenciado com as imagens 26 e 27, a qual a primeira representa não só o elemento filho de unidade de apresentação sobrepondo o elemento pai, a unidade de apresentação, mas também demonstrando que ao ser arrastado os elementos internos seguem proporcionalmente.

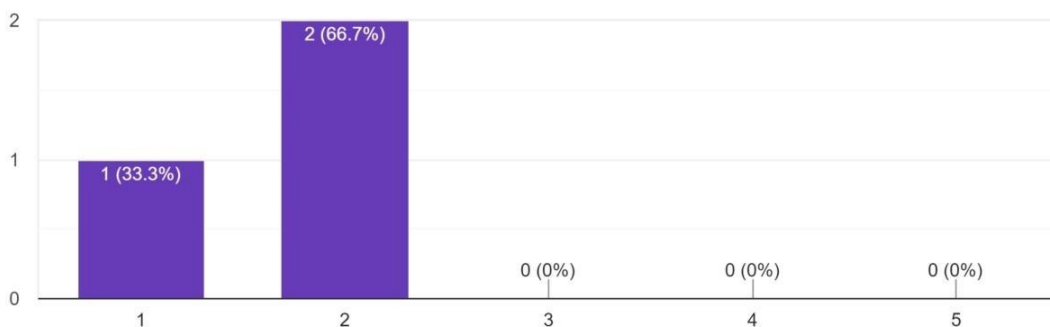
Ademais, é válido ressaltar que quando o projeto atingiu maturidade, com as validações dos conectores e o desenvolvimento da notação completa, o código foi disponibilizado e apresentado para o time de desenvolvimento, vide que eles serão os responsáveis por continuar o desenvolvimento.

Assim, entendendo que a experiência do desenvolvedor também foi levada em consideração para o estudo. Com objetivo de obter resultados iniciais sobre a satisfação com a nova ferramenta foi elaborado um questionário distribuído entre os desenvolvedores do projeto o desenvolvimento do questionário foi utilizado o Google Formulário e as questões aprovadas pela professora orientadora, com um total de 3 respostas dos 6 desenvolvedores ativos no projeto. Assim segue abaixo as perguntas e os resultados obtidos com os devidos comentários.

Figura - 28 Gráfico e pergunta um sobre experiência com o código refatorado

Como você avalia maior facilidade de uso do React Flow em comparação com o mxgraph?

3 respostas



Fonte: Elaborado pelo autor (2024).

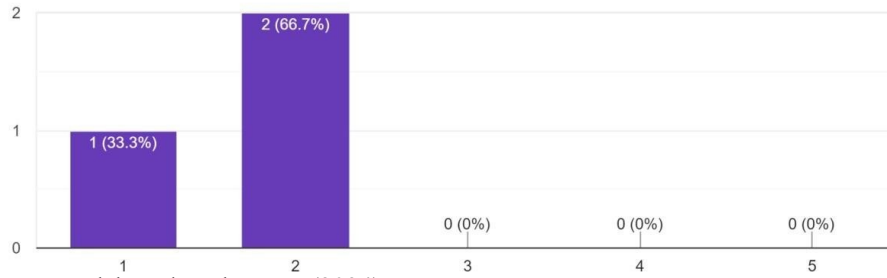


A figura 28 representa a uma pergunta em que 1 seria concordo plenamente e 5 seria discordo plenamente.

Figura - 29 Gráfico e pergunta dois sobre o código refatorado

Você acredita que com essa mudança de biblioteca para Reactflow poderá ajudar na manutenção do código?

3 respostas



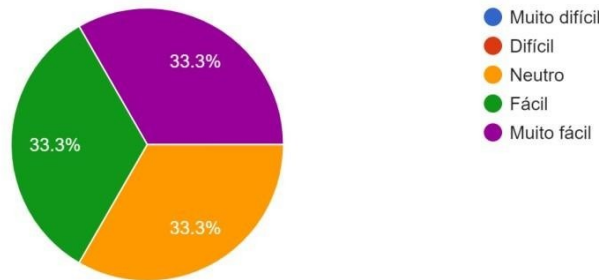
Fonte: Elaborado pelo autor (2024).

A figura 29 representa a uma pergunta em que 1 seria concordo plenamente e 5 seria discordo plenamente.

Figura - 30 Gráfico e pergunta três sobre o código refatorado

O quanto a documentação do React Flow pode facilitar durante o processo de novas implementações ?

3 respostas

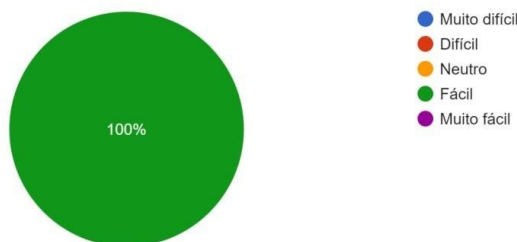


Fonte: Elaborado pelo autor (2024).

Figura - 31 Gráfico e pergunta quatro sobre o código refatorado

Quão intuitivo seria aprender e utilizar as funcionalidades do React Flow em comparação com o mxgraph?

3 respostas



Fonte: Elaborado pelo autor (2024).

Assim, apesar da baixa amostra de participantes, pode-se concluir que a refatoração foi demonstra aspectos positivos para os desenvolvedores, principalmente ao comparar as notações mxgraph e o código refatorado com reactflow.

## 7 CONCLUSÃO E TRABALHOS FUTUROS

A refatoração dos recursos principais da USINN Modeler trouxe melhorias substanciais tanto em clareza do código quanto na manutenção, aspectos críticos para a evolução sustentável do projeto. A adoção do React Flow como principal ferramenta para criação de diagramas, aliada a práticas de código limpo, resultou em uma organização mais eficiente e uma redução de duplicidades, permitindo que o sistema seja mais fácil de compreender e modificar. A nova abordagem de validação de conexões entre elementos e a escolha de uma biblioteca amplamente suportada aumentaram a confiança na escalabilidade e na continuidade do projeto, proporcionando uma experiência de desenvolvimento mais fluida e intuitiva.

Para trabalhos futuros, é essencial explorar mais a fundo a experiência do desenvolvedor, realizando testes com um público mais amplo que vá além da equipe de QA do projeto. Isso permitirá uma avaliação mais detalhada das melhorias introduzidas e ajudará a identificar potenciais gargalos que possam ainda existir. Testes de regressão na versão atual e da nova são de suma importância pois com ele será possível identificar possíveis bugs na atual e comparar com a nova se já foram corrigidos e também é possível verificar se não existe funcionalidades faltando a ferramenta refatorada. Testes de usabilidade com usuários externos, especialmente aqueles que não estão familiarizados com a ferramenta, serão cruciais para validar a eficiência das mudanças feitas. Além disso, uma pesquisa mais aprofundada sobre a integração, feedback contínuo dos usuários e desenvolvimento de recursos de ajuda e tutorial poderá orientar futuras evoluções da plataforma, garantindo que o USINN Modeler atenda às necessidades tanto do público acadêmico quanto de profissionais da área de design de interação.

## REFERÊNCIAS

- MARLI, A.E.D.A. (1983). **Texto, contexto e significado: algumas questões na análise de dados qualitativos**. Cadernos de Pesquisa, (45): 66-71.
- ASCENCIO, A.F.G. (2000) **Método Heurístico para Projeto de Interfaces Inteligentes com Usabilidade** (Dissertação de Mestrado), Universidade Federal do Rio Grande do Sul.
- BECK, K. **Extreme Programming Explained: Embrace Change**. Addison-Wesley, 2000.
- BEVAN, N. (1995) **Usability is quality of use**. In: Anzai & Ogawa (eds) Proc. 6th International Conference on Human Computer Interaction, July. Elsevier.  
<http://www.usability.serco.com/papers/usabis95.pdf>.
- BREZOLIN, Carmen Vera Scorsatto; SILVEIRA, Milene Selbach. **Percepções de Professores sobre Recursos de Ajuda em Ferramentas de Programação baseadas em Blocos**. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO (SBIE), 33. , 2022, Manaus. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2022 . p. 1281-1291. DOI: <https://doi.org/10.5753/sbie.2022.224718>.
- D. GRAZIOTIN, F. Fagerholm, X. Wang, and P. Abrahamsson, **On the unhappiness of software developers**, in Proceedings of the 21st international conference on evaluation and assessment in software engineering, 2017.
- FEITOSA, Victor; SANTOS, Alex Alan; FIORI, Maria Victoria; MARQUES, Anna Beatriz. **Design de interação com a notação USINN: resultados iniciais sobre a experiência prática de designers**. In: PÔSTERES E DEMONSTRAÇÕES - SIMPÓSIO BRASILEIRO DE FATORES HUMANOS EM SISTEMAS COMPUTACIONAIS (IHC), 21. , 2022, Diamantina. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2022 . p. 128-131. DOI: [https://doi.org/10.5753/ihc\\_estendido.2022.226190](https://doi.org/10.5753/ihc_estendido.2022.226190).
- FOWLER, M. **Refactoring: Improving the Design of Existing Code**. Addison-Wesley, 1999.
- GIBBS, Graham. **Análise de dados qualitativos** / Graham Gibbs; tradução Roberto Cataldo Costa; consultoria, supervisão e revisão técnica desta edição Lorí Viali. - Dados eletrônicos. - Porto Alegre: Artmed, 2009. (Coleção Pesquisa qualitativa / coordenada por Uwe Flick).
- GOLDENBERG, J., & Mazursky, D. (2002). **Creativity in Product Innovation**. Cambridge University Press.
- HERRMANN, M. e Silveira, M.S. (2005) **“Sistemas de Ajuda Online em Softwares Educacionais: algumas considerações e percepções”**. In: Anais do XVI Simpósio Brasileiro de Informática na Educação, p. 477-485.

HEZORG, A.; Shahmehri, N. **“User help techniques for usable security.”** In: CHIMIT '07: Proceedings of the 2007 Symposium on Computer Human Interaction for the Management of Information Technology, 2007, pp. 635- 642.

HUANG, J.; Lu, B.; Michael B. **“Graphical abstract help”**. In: CHINZ '05 Proceedings of the 6th ACM SIGCHI New Zealand Chapter's International Conference on Computer-Human Interaction: Making CHI Natural, 2005, pp. 83-89.

MARQUES, A. B.; CONTE, T.; DINIZ, S. 2016. **Representing the interaction and navigation of interactive systems through a usability-oriented model: A feasibility study.** In Proceedings of the 15th Brazilian Symposium on Human Factors in Computing Systems. Association for Computing Machinery, New York, NY, USA, Article 15, 1–10.

MARQUES, A. B.; BARBOSA, S. D. J.; CONTE, T. **Exploring how experienced and unexperienced designers use and evaluate a usability-oriented interaction and navigation model.** Journal on Interactive Systems, Porto Alegre, RS, v. 9, n. 3, 2018. DOI: 10.5753/jis.2018.709. Disponível em: <https://sol.sbc.org.br/journals/index.php/jis/article/view/709>. Acesso em: 20 nov. 2023.

Marques A (2024). **Exploring novice designers acceptance of the use of a tool for modeling the interaction and usability of information systems.** In Proceedings of the 20th Brazilian Symposium on Information Systems (SBSI '24). Association for Computing Machinery, New York, NY, USA, Article 45, 1–10. <https://doi.org/10.1145/3658271.3658316>

MARTIN, R. C. **Clean Code: A Handbook of Agile Software Craftsmanship.** Prentice Hall, 2009.

MILES, M. B., Huberman, A. M., & Saldaña, J. (2014). **Qualitative data analysis: A methods sourcebook.** Thousand Oaks, CA: SAGE Publications.

NIELSEN, J. (1993). **Usability Engineering.** Academic Press.

NIELSEN, J. **Usability engineering.** Morgan Kaufmann, 1994. ISBN 0-12-518406-9

NORMAN, Kent L. **The Wiley Handbook of Human Computer Interaction Set,** 2018.

PELLISSARI, Estelamaris. **Estruturação e apresentação de sistemas de ajuda on-line para software educacional.** 2009. 137 f. Dissertação (Mestrado em Ciência da Computação) - Pontifícia Universidade Católica do Rio Grande do Sul, Porto Alegre, 2009.

PERLMAN, G. **Coordinating consistency of user interfaces, code, online help, and documentation with multilingual/multitarget software specification,** 1989.

POLTROCK, S. E. **Participant-observer studies of user interface design and development,** 1994.

ROSENBAUM, S.; Kantner, L.; Dworman, G. **“Helping Users to Use Help: Results from Two International Conference Workshops”**. In: IEEE International Professional Communication Conference Proceedings, 2005, pp. 1717-1718

SHNEIDERMAN, B. (1998). **Designing the User Interface: Strategies for Effective Human-Computer Interaction**. Addison-Wesley.

SIMONE D. J. Barbosa e Máira G. de Paula. **Designing and Evaluating Interaction as Conversation: a Modeling Language based on Semiotic Engineering**, 2003.

SOMMERVILLE, Ian.; tradução Ivan Bosnic e Kalinka G. de O. Gonçalves; revisão técnica Kechi Hiram. **Engenharia de Software** / — 9. ed. — São Paulo: Pearson Prentice Hall, 2011. Título original: Software engineering. ISBN 978-85-7936-108-1

VOULIGNY, L.; Robert, J. **“Online help system design based on the situated action theory”**. In: CLIHC '05 Proceedings of the 2005 Latin American Conference on Human- Computer Interaction, 2005, pp. 64-75.

RAYMOND, E. S. *The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary*. O'Reilly Media, 2001.