



**UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO COLEGIADO DO CURSO
CIÊNCIA DA COMPUTAÇÃO**

Vinícius de Oliveira Risso

**APLICAÇÃO DE UMA ABORDAGEM BASEADA EM
ONTOLOGIA NA EVOLUÇÃO DE UM SISTEMA COM
INTERFACE ADAPTATIVA**

VITÓRIA, ES

2023

Vinícius de Oliveira Risso

**APLICAÇÃO DE UMA ABORDAGEM BASEADA EM ONTOLOGIA
NA EVOLUÇÃO DE UM SISTEMA COM INTERFACE
ADAPTATIVA**

Monografia apresentada ao Curso de Ciência da Computação do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do Grau de Bacharel em Ciência da Computação.

Universidade Federal do Espírito Santo

Centro Tecnológico

Colegiado do Curso Ciência da Computação

Orientadora: Monalessa Perini Barcellos

Coorientador: Alexandre Adler Cunha de Freitas

VITÓRIA, ES

2023

Vinícius de Oliveira Risso

Aplicação de uma Abordagem Baseada em Ontologia na Evolução de um Sistema com Interface Adaptativa/Vinícius de Oliveira Risso. – Vitória, ES, 2023 - [69](#) p. : il. (algumas color.) ; 30 cm.

Orientador: Monalessa Perini Barcellos

Monografia (PG) – Universidade Federal do Espírito Santo – UFES
Centro Tecnológico
Colegiado do Curso de Ciência da Computação, 2023.

1. Interface Adaptativa. 2. Ontologia. 3. Abordagem 4. Sistema Adaptativo . I. Risso, Vinícius de Oliveira. II. Universidade Federal do Espírito Santo. IV. Aplicação de uma Abordagem Baseada em Ontologia na Evolução de um Sistema com Interface Adaptativa.

CDU 02:141:005.7

Vinicius de Oliveira Risso

**APLICAÇÃO DE UMA ABORDAGEM BASEADA EM
ONTOLOGIA NA EVOLUÇÃO DE UM SISTEMA COM
INTERFACE S ADAPTATIVA**

Monografia apresentada ao Curso de Ciência da Computação do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do Grau de Bacharel em Ciência da Computação.

Trabalho aprovado. Vitória, ES, 20 de julho de 2023:

Profa. Monalessa Perini Barcellos, D. Sc.
Orientadora

Alexandre Adler Cunha de Freitas, M.Sc.
Coorientador

Prof. Vítor E. Silva Souza, Ph. D.
Universidade Federal do Espírito Santo

Profa. Simone Dornelas Costa, D. Sc.
Universidade Federal do Espírito Santo

VITÓRIA, ES

2023

“Dedico este trabalho a todos os que me
ajudaram ao longo desta caminhada.”

AGRADECIMENTOS

Gostaria de expressar minha gratidão a Deus por ter me concedido saúde e condições físicas e mentais que me permitiram concluir minha graduação.

Agradeço também a todos os que marcaram minha vida, dentro e fora da universidade, que me fizeram refletir, aprimorar e desenvolver meu senso crítico e espiritual. Aos que me apoiaram, ajudaram, me ouviram quando eu precisei e me aconselharam quando necessário. De maneira especial, gostaria de agradecer especificamente às pessoas em que pude confiar a todo momento, que estiveram ao meu lado e me ergueram sempre que caí e fraquejei. Gostaria de agradecer especialmente aos meus pais, que sempre estiveram ao meu lado e me ajudaram a permanecer no caminho certo.

Gostaria de expressar meus sinceros agradecimentos a meus pais, Valcimar José Risso e Rozely de Oliveira Risso, por serem um porto de sabedoria e paciência em minha vida. Devo também fazer menção honrosa a muitas pessoas que estiveram nesta caminhada longa que chegou ao fim, gostaria de agradecer aos meus colegas de curso e amigos que me apoiaram e ajudaram a seguir em frente na vida: Franco Schmidt Rossi, Rafael Franco, Bruno Gama, Gustavo Alocchio, João Reis Ubaldo, Christiane Garcia (que desenvolveu a Figura 20) dentre vários outros, com certeza, sem vocês, teria sido muito mais difícil o processo.

Não possuo palavras suficientes para agradecer ao meu amigo Franco, um dos amigos mais importantes que fiz na UFES, um parceiro em que fazer trabalhos acadêmicos e estudar para provas não era o limite. A confiança e a convivência que compartilhamos tornaram nossa amizade incrível.

Por fim, devo agradecer a todos os professores do Curso de Ciências da Computação da Universidade Federal do Espírito Santo (UFES), por compartilhar e ensinar tudo que aprendi ao longo do curso. De maneira especial, gostaria de agradecer aos professores Vítor Estevão e Simone Dornelas, por aceitarem participar da minha banca neste momento único da minha graduação. Obrigado por fazerem parte deste momento tão importante para mim. Ao meu Coorientador Alexandre Adler Cunha de Freitas, por ser meu coorientador, mentor e amigo. Minha orientadora, Monalessa, o empenho destas duas últimas menções é nítido e tem um lugar guardado em minhas lembranças.

RESUMO

Com a popularização dos computadores pessoais e *smartphones*, novas formas de interação com dispositivos eletrônicos surgiram. Isso impulsionou o desenvolvimento de um novo campo de estudo chamado Interação Humano-Computador (IHC). Um problema comum em IHC é a baixa usabilidade quando diferentes usuários utilizam o mesmo sistema. Uma solução para esse problema são os sistemas adaptativos, que podem ajustar aspectos de sua estrutura ou funcionalidades para melhor se adequar às necessidades dos diferentes usuários e suas alterações ao longo do tempo. IHC é um campo interdisciplinar que envolve diversos especialistas e áreas de conhecimento, o que pode levar a uma falta de consenso em relação a conceitos e terminologias. Isso pode gerar conflitos semânticos na compreensão e modelagem das características do usuário, entre outros aspectos. Este trabalho propõe a utilização de ontologias no processo de desenvolvimento e evolução de sistemas adaptativos, particularmente aqueles cuja adaptabilidade inclui sua interface. Ontologias têm sido amplamente empregadas em diversos campos para capturar e estruturar conhecimento, com o objetivo de solucionar problemas relacionados à interoperabilidade e à gestão de informações e conhecimento. Neste trabalho, utilizando a abordagem OADAPT (*Ontology-based Approach to Develop Adaptive Interfaces*) foram propostas e implementadas novas adaptações na interface do sistema SNOPI, uma rede social com interface adaptativa baseada em ontologia desenvolvido em um trabalho anterior. Foram implementadas duas novas adaptações na interface adaptativa do SNOPI, incluindo a navegação e postagem no sistema por meio de comandos de voz e a navegação por gestos das mãos e dedos. Como resultado, SNOPI foi aprimorado com essas novas adaptações, que foram projetadas e implementadas com sucesso.

Palavras-chaves: Interface Adaptativa, Ontologia, Abordagem, Sistema Adaptativo.

LISTA DE FIGURAS

Figura 1 - Extrato de HCI-ON. Adaptado de Costa (2022).	9
Figura 2 - Fragmento de HCI-ON utilizado no desenvolvimento de SNOPI. Adaptado de (FREITAS et al., 2022).	11
Figura 3 - Tela de Login e Cadastro de Usuário (SCALSER, 2022).	14
Figura 4 - Questionário exibido ao usuário após seu primeiro login (SCALSER, 2022).	15
Figura 5 - Tela de Feed do sistema com adaptação - Modo escuro e fonte de letra aumentada (SCALSER, 2022).	16
Figura 6 - Visão geral do processo baseado em ontologia para desenvolver sistemas com interfaces adaptativas (FREITAS et al., 2022).	17
Figura 7 - Mapa de Empatia baseado na Persona Alex.	25
Figura 8 - Fragmento de HCI-ON incluindo novos conceitos.	28
Figura 9 - Diagrama de Classes de SNOPI2.0.	31
Figura 10 - Diagrama da arquitetura de software do SNOPI (SCALSER, 2022).	32
Figura 11 - Ontologia Operacional - hierarquia de classes definida.	35
Figura 12 - Axioma da Classe Hand_Free_Mode	38
Figura 13 - Axioma da Classe Voice_Command_Mode	38
Figura 14 - Propriedades do indivíduo '01'.	39
Figura 15 - Inferência sobre o indivíduo '01'.	39
Figura 16 - Representação gráfica da ontologia operacional.	40
Figura 17 - Fragmento do código utilizado para manipulação da ontologia operacional (SCALSER, 2022).	41
Figura 18 - Fragmento do código do módulo de navegação por gestos (Mãos Livres).	43
Figura 19 - Fragmento do código do módulo de navegação/postagens por comandos de voz.	44
Figuras 20 - Diagramação das adaptações.	46
Figura 21 - Modal de pesquisa do perfil do usuário com novas perguntas das novas adaptações.	46
Figura 22 - Funcionalidades e equivalências visuais por gestos.	48
Figura 23 - Print do sistema usando o gesto 'mão fechada'.	49
Figura 24 - Print do sistema usando o gesto 'mão aberta'.	49
Figura 25 - Funcionalidades e equivalências visuais por voz.	50
Figura 26 - Print do sistema - adaptação comando de voz.	51

LISTA DE TABELAS

Tabela 1 - Persona Alex - Usuário com Lesão por Esforço Repetitivo (LER) (ABOU-ZAHRA; SINCLAIR, 2017).	24
Tabela 2 - Novas adaptações definidas.	26
Tabela 3 - Requisito funcional ajustado.	27
Tabela 4 - Novos requisitos não funcionais identificados.	27
Tabela 5 - Novas perguntas incluídas no questionário do usuário.	33
Tabela 6 - Classes de Subclasses adicionadas a ontologia ontoSNOPI2.0.	36
Tabela 7 - Legenda dos comandos.	47
Tabela 8 - Objetivos e seus resultados.	54

LISTA DE ABREVIATURAS

AIO	Adaptive Interface Ontology
API	Application Programming Interface
DCO	Device Context Ontology
EM	Esclerose Múltipla
CSS	Cascading Style Sheets
HTML	HyperText Markup Language
HCI-ON	Human-Computer Interaction Ontology Network
IHC	Interação Humano-Computador
LER	Lesão por Esforço Repetitivo
NEMO	Núcleo de Estudos em Modelagem Conceitual e Ontologias
RSI	Repetitive Stress Injury
W3C	World Wide Web Consortium
MDD	Model-Driven Development
MVC	Model-View-Controller
MVVM	Model-View-ViewModel
SCSS	Sassy Cascading Style Sheets
SEON	Software Engineering Ontology Network
SNOPI	Social Network with Ontology-based Adaptive Interface
SysSwO	System and Software Ontology
SWRL	Semantic Web Rule Language
OADAPT	Ontology-based Approach to Develop Adaptive Interfaces
OBA	Ontology-Based Architectures
ODD	Ontology-Driven Development
OWL	Web Ontology Language
SCSS	Sassy Cascading Style Sheets
SWRL	Semantic Web Rule Language
UCO	User Characterization Ontology
UFO	Unified Foundational Ontology
UIT&EO	UI Types and Elements Ontology
WAI	Web Accessibility Initiative

Sumário

Capítulo 1	1
Introdução	1
1.1 Introdução	1
1.2 Objetivos do Trabalho	3
1.3 Método de Desenvolvimento	3
1.4 Organização do Texto	4
Capítulo 2	6
Fundamentação Teórica	6
2.1 Interfaces Adaptativas	6
2.2 Ontologias	7
2.3 Uso de Ontologias no Desenvolvimento de Interfaces do Usuário Adaptativas	12
2.3.1 SNOPI (Social Network with Ontology-based adaPtive Interface)	13
2.3.2 OADAPT (Ontology-based Approach to Develop AdaPtive inTerfaces)	16
2.4 Tecnologias Utilizadas neste Trabalho	20
2.5 Considerações Finais do Capítulo	21
Capítulo 3	23
Aplicação da Abordagem OADAPT para Evolução do SNOPI	23
3.1 Identificar o Escopo e os Usuários do Sistema	23
3.1.1 Identificação dos Usuários do Sistema e suas Características	23
3.1.2 Identificação de Novas Adaptações	26
3.2 Elicitar Requisitos do Sistema	27
3.3 Selecionar Ontologia	28
3.4 Realizar Análise do Sistema	29
3.4.1 Diagrama de Classes	29
3.5 Definir a Arquitetura do Sistema	32
3.6 Definir Adaptações de Interface do Usuário	32
3.7 Desenvolver Ontologia Operacional	34
3.7.1 Evolução da OntoSNOPI2.0	34
3.7.2 Manipulação e Reasoning da OntoSNOPI2.0	41
3.8 Implementar e Testar o Sistema	42
3.9 Considerações Finais do Capítulo	44
Capítulo 4	45
SNOPI 2.0	45
4.1 SNOPI 2.0	45
4.2 Adaptação Mãos Livres	47
4.3 Adaptação Comando de Voz	49
4.4 Considerações Finais do Capítulo	51
Capítulo 5	52
Considerações Finais	52
5.1 Conclusões	52
5.2 Limitações	54
5.3 Trabalhos Futuros	55
Referências Bibliográficas	57

Capítulo 1

Introdução

Este capítulo apresenta uma breve introdução ao tema do trabalho, seus objetivos, histórico de desenvolvimento e a organização deste documento.

1.1 Introdução

Desde a criação e democratização de computadores pessoais e *smartphones*, novas formas de manusear dispositivos foram surgindo. Com isso, também surgiu um novo ramo de estudos, conhecido atualmente como Interação Humano-Computador (IHC). Um problema relacionado a IHC conhecido e muito discutido atualmente é a baixa usabilidade quando diferentes tipos de usuários utilizam o mesmo sistema. Uma solução possível para esse problema são os sistemas adaptativos, pois eles conseguem alterar aspectos de sua estrutura ou funcionalidades e, com isso, acomodar e atender melhor às necessidades de diferentes usuários e suas mudanças ao longo do tempo (SCALSER, 2022).

Um sistema adaptativo é capaz de alterar automaticamente suas características e funcionalidades de acordo com as necessidades dos usuários, possibilitando que o sistema se adapte e melhore continuamente ao longo do tempo, tornando a experiência do usuário mais eficiente e intuitiva (OPPERMANN, 1994). Dentre as adaptações que podem ser realizadas em um sistema adaptativo estão as adaptações na interface do usuário. O uso de interfaces adaptativas possibilita que os sistemas sejam capazes de adaptar sua interface ao comportamento e necessidades dos usuários, tornando-os mais fáceis de serem utilizados e aumentando sua satisfação (NIELSEN, 1994).

As interfaces adaptativas podem melhorar a acessibilidade do sistema, visto que, ao se adaptarem automaticamente às necessidades de usuários com algum tipo de deficiência ou limitação, tornam o sistema mais acessível e inclusivo, abrangendo vários públicos (ARAÚJO, 2017). No entanto, desenvolver interfaces adaptativas não é trivial. Esse processo envolve não apenas atividades relacionadas ao desenvolvimento de software, mas também conhecimento acerca de sistemas adaptativos (suas características, seus componentes, etc.), do perfil dos usuários, das adaptações adequadas para cada usuário, entre outros.

Ontologias podem ajudar nesse contexto. Ontologias são úteis para representar conhecimento de maneira explícita e formal e, assim, podem auxiliar provendo conhecimento útil para a concepção da interface, bem como servindo como um artefato computacional que auxilia na sua implementação.

Ontologias vêm sendo utilizadas para apoiar o desenvolvimento de sistemas. Por exemplo, ontologias podem ser usadas em *Ontology-Driven Development* (ODD), ou seja, quando são aplicadas em tempo de desenvolvimento para descrever o domínio do problema (abordagens orientadas a modelos, como por exemplo, Desenvolvimento Orientado a Modelos (*Model-Driven Development* – MDD)), ou, em *Ontology-Based Architectures* (OBA), quando as ontologias são utilizadas como artefatos principais em tempo de execução e compõem uma parte central da lógica da aplicação, como em abordagens de regras de negócios, elas podem proporcionar uma representação precisa e consistente dos conceitos e relações presentes no domínio de aplicação (HAPPEL e SEEDORF, 2006). Sendo assim, o estudo sobre o desenvolvimento de sistemas com interfaces adaptativas utilizando ontologias é importante para contribuir para a criação de sistemas mais eficazes, eficientes e intuitivos, que atendam às necessidades dos diferentes tipos de usuários.

Levando em consideração os benefícios providos por um sistema com interface adaptativa, como, por exemplo, uma melhor interação com o usuário de acordo com suas características, e o uso bem sucedido de ontologias no desenvolvimento de sistemas de software em geral (entre eles, sistemas adaptativos), encontra-se em desenvolvimento no NEMO¹ (Núcleo de Estudos em Modelagem Conceitual e Ontologias), grupo de pesquisa no qual este trabalho está sendo realizado, uma pesquisa de Doutorado, cujo autor é o coorientador deste trabalho, que investiga o uso de ontologias no desenvolvimento de interfaces adaptativas.

No contexto dessa pesquisa, foi realizado um estudo exploratório (FREITAS *et al.*, 2022) no qual foi desenvolvida a ferramenta SNOPI² (*Social Network with Ontology-based adaptive Interface*) (SCALSER, 2022), que utiliza ontologias de HCI-ON (*Human-Computer Interaction Ontology Network*) (COSTA *et al.*, 2020), tanto em ODD quanto em OBA, para adaptar a interface de uma rede social de acordo com limitações relacionadas à visão de seus usuários. Com base na experiência desse estudo, foi definida a abordagem OADAPT (*Ontology-based Approach to Develop Adaptive Interfaces*)

¹ <https://nemo.inf.ufes.br/>

² <https://dev.nemo.inf.ufes.br/snopi/>

(FREITAS *et al.*, 2022), que propõe um conjunto de passos para apoiar o desenvolvimento de interfaces adaptativas baseadas em ontologia.

Considerando as limitações atuais de SNOPI, particularmente seu conjunto limitado de adaptações, surge a oportunidade de evoluí-la. O desenvolvimento de novas adaptações e melhorias em SNOPI é relevante, pois, além dos benefícios oferecidos pelo sistema em si, será possível colaborar com pesquisas em andamento no NEMO, que visam solucionar problemas semânticos e de representação de conhecimento enfrentados pelos desenvolvedores de sistemas interativos, particularmente no diz respeito à melhoria da usabilidade dos sistemas.

Para desenvolver novas adaptações e melhorias em SNOPI foi utilizada a abordagem OADAPT (FREITAS *et al.*, 2022). Dessa forma, este trabalho serve como um estudo de caso de aplicação de OADAPT e contribui para avaliá-la (*e.g.*, observando se seu uso auxilia no desenvolvimento do sistema e contribui para o desenvolvimento de uma interface capaz de se adaptar de maneira eficiente e eficaz às mudanças no contexto de uso e nas necessidades dos usuários) e melhorá-la.

1.2 Objetivos do Trabalho

Este trabalho tem como objetivo geral propor melhorias nas adaptações de interface no sistema de interface adaptativa SNOPI (*Social Network with Ontology-based adaptive Interface*) utilizando OADAPT (*Ontology-based Approach to Develop Adaptive Interfaces*). Esse objetivo geral pode ser detalhado nos seguintes objetivos específicos:

- i. Especificar a evolução nas adaptações a serem realizadas em SNOPI;
- ii. Projetar a evolução de SNOPI;
- iii. Implementar e testar a evolução projetada;
- iv. Fornecer feedback acerca do uso de OADAPT.

1.3 Método de Desenvolvimento

Este trabalho foi conduzido de acordo com as seguintes etapas:

- i. **Revisão da Literatura:** o trabalho foi iniciado com uma revisão bibliográfica sobre interfaces adaptativas em IHC, ontologias e redes ontologias, na qual foram lidos materiais (livros, teses, dissertações e artigos científicos) no contexto do assunto;

- ii. **Estudo do SNOPI e suas tecnologias:** consistiu no estudo de SNOPI para conhecer o sistema e as tecnologias utilizadas em seu desenvolvimento;
- iii. **Identificação da evolução a ser realizada:** consistiu na identificação das novas adaptações que o sistema deve ser capaz de realizar. Nesta etapa, foram realizadas as duas primeiras etapas de OADAPT, que consistiram em *Identificar o escopo e os usuários do sistema* e *Elicitar requisitos do sistema*;
- v. **Desenvolvimento de nova versão de SNOPI:** consistiu em projetar, implementar e testar as novas adaptações da solução. Nesta etapa foram realizadas as demais atividades de OADAPT, que consistiram em *Selecionar ontologia*, *Realizar análise do sistema*, *Definir arquitetura do sistema*, *Definir adaptações da Interface do usuário*, *Desenvolver ontologia operacional* e *Implementar e Testar o sistema*;
- vi. **Escrita da Monografia:** consistiu na escrita deste documento.

1.4 Organização do Texto

Este primeiro capítulo contém a introdução do trabalho, assim como, seus objetivos e suas etapas de desenvolvimento. Além deste capítulo, a monografia possui outros 4 capítulos, sendo eles:

1. **Capítulo 2 – Fundamentação Teórica:** apresenta uma breve fundamentação teórica sobre Interfaces Adaptativas, Ontologias e o Uso de Ontologias no Desenvolvimento de Interfaces de Usuário Adaptativas, o sistema SNOPI (*Social Network with Ontology-based adaptive Interface*) e a abordagem OADAPT (*Ontology-based Approach to Develop Adaptive Interfaces*). Também apresenta as tecnologias que foram utilizadas no desenvolvimento do sistema.
2. **Capítulo 3 – Aplicação da Abordagem OADAPT para Evolução do SNOPI:** apresenta em detalhes as etapas seguidas para a implementação de novas adaptações no sistema SNOPI, são elas: (i) Identificar o escopo e os usuários do sistema; (ii) Elicitar requisitos do sistema; (iii) Selecionar ontologia; (iv) Realizar análise do sistema; (v) Definir a arquitetura do sistema; (vi) Definir adaptações de interface do usuário; (vii) Desenvolver ontologia operacional; e (viii) Implementar e testar o sistema.
3. **Capítulo 4 – SNOPI 2.0:** apresenta as novas adaptações implementadas no sistema SNOPI.

4. [Capítulo 5](#) – **Considerações finais:** apresenta as considerações finais do trabalho, incluindo algumas dificuldades encontradas, limitações do sistema evoluído e experiências adquiridas. Além disso, são feitas algumas sugestões para trabalhos futuros.

Capítulo 2

Fundamentação Teórica

Neste capítulo são apresentados os principais fundamentos teóricos relacionados a este trabalho. Na [Seção 2.1](#) são tratados aspectos sobre Interfaces Adaptativas. A [Seção 2.2](#) aborda Ontologias. Na [Seção 2.3](#) são apresentados alguns trabalhos relacionados ao uso de ontologias no desenvolvimento de interfaces adaptativas e são apresentados SNOPI e OADAPT. Na [Seção 2.4](#) são abordadas as tecnologias utilizadas neste trabalho. Por fim, na [Seção 2.5](#) são feitas as considerações finais do capítulo.

2.1 Interfaces Adaptativas

Antes de definir o que é uma interface adaptativa, é preciso entender o que é uma interface com o usuário. Uma interface com o usuário é a maneira como duas entidades se comunicam. Ela é como uma fronteira que define o que cada entidade pode fazer e como elas podem se comunicar (SCAPIN, 1990).

Interfaces adaptativas caracterizam um tipo particular de sistema adaptativo (BENYON, 1988). Considerando-se que a característica principal de um sistema adaptativo é ser capaz de mudar características e funcionalidades de maneira automática de acordo com as necessidades do usuário (OPPERMANN, 1994), quando essas adaptações são realizadas na interface com o usuário, tem-se uma interface adaptativa.

Outras duas definições para interface adaptativa são:

- Uma interface com o usuário adaptativa é um artefato de software que melhora sua capacidade de interagir com um usuário, construindo um modelo do usuário com base na experiência parcial do mesmo (LANGLEY, 1999).
- Uma interface do usuário adaptativa é uma interface com o usuário que se adapta, ou seja, muda seu *layout* e elementos às necessidades do usuário ou contexto e pode ser alterado de forma semelhante por cada usuário (SCHNEIDER-HUFSCHMIDT, MALINOWSKI, 1993).

É importante deixar claro que uma interface adaptativa é um conceito diferente de interface adaptável. Enquanto uma interface adaptável fornece uma opção de

alteração pelo usuário na interface, uma interface adaptativa fornece modificações automáticas de acordo com as necessidades do usuário (JAMESON, 2007).

Interfaces com o usuário podem ser alteradas em diversos aspectos, como o design (layout do sistema), o tamanho dos elementos, o contraste de cores (*e.g.*, em imagens, planos de fundo, fontes, entre outros). Essas alterações permitem que a interface com o usuário seja ajustada de acordo com as necessidades e preferências do usuário, tornando a interação mais eficaz e intuitiva.

2.2 Ontologias

Uma ontologia é um modelo conceitual que descreve uma realidade. Segundo Guizzardi (2007), uma ontologia é uma especificação de um metamodelo³ de uma linguagem para representar fenômenos de um determinado domínio da realidade. Em outras palavras, trata-se de uma linguagem que só admite que sejam especificados possíveis estados de coisas que representem a realidade.

Segundo Scherp (2011), as ontologias podem ser organizadas em três camadas arquiteturais, sendo elas:

(i) *Ontologias de fundamentação*, que possuem um escopo mais geral. São amplamente reutilizadas em diferentes cenários de modelagem e visam representar conceitos básicos e gerais do mundo, bem como as relações entre esses conceitos, como objetos e eventos;

(ii) *Ontologias de núcleo*, que são um refinamento de ontologias de fundamentação e, com isso, proporcionam uma descrição mais precisa, mas abstrata, da estruturação do conhecimento em um contexto de modelagem, como por exemplo, serviços e organizações. São genéricas e abrangem um conjunto de subdomínios;

(iii) *Ontologias de domínio*, nas quais encontra-se a representação do conhecimento que é específico para um determinado domínio, como por exemplo, o desenvolvimento de software ou a gestão de uma universidade.

Uma outra forma de classificar ontologias é realizando a distinção entre as *ontologias de referência* (modelo conceitual) e as *ontologias operacionais* (artefatos computacionais) (GUIZZARDI, 2007). As ontologias de referência têm como objetivo descrever com maior fidelidade possível a conceituação para qual ela foi

³ O metamodelo é um modelo utilizado para descrever outros modelos. Ele fornece uma estrutura geral para entender como outros modelos funcionam e como eles se relacionam uns com os outros (CHEN; DING; JIN, 2010).

idealizada (GUIZZARDI, 2007). De maneira contrária às ontologias de referência, as ontologias operacionais buscam garantir que propriedades computacionais sejam satisfeitas (FALBO *et al.*, 2013).

A organização do conhecimento em domínios grandes e complexos é um desafio. Uma abordagem comum é a criação de uma única ontologia para representar todo o domínio, mas isso pode ser custoso devido à dificuldade de manipulação, criação, uso, reúso e manutenção. Por outro lado, representar cada subdomínio isoladamente também é uma tarefa custosa, devido ao seu tamanho e às dificuldades relacionadas à integração e alinhamento de ontologias (RUY *et al.*, 2016). Nesse sentido, uma estratégia para facilitar o desenvolvimento/uso dessas ontologias é a organização em redes de ontologias. Uma rede de ontologias é composta por uma coleção de ontologias relacionadas entre si por meio de relações como dependência e alinhamento (SUÁREZ-FIGUEROA, 2012).

O domínio da IHC apresenta uma ampla variedade de ontologias, muitas vezes utilizadas de forma isolada, apesar da existência de conceitos comuns entre elas (COSTA, 2022). Isso dificulta ter uma compreensão abrangente do domínio, bem como a reutilização da conceituação como um todo ou em partes. A partir desta problemática, COSTA *et al.* (2020) propuseram a HCI-ON⁴ (*Human-Computer Interaction Ontology Network*), uma rede de ontologias que contém ontologias que tratam aspectos relacionados a IHC. A [Figura 1](#) apresenta uma visão geral de HCI-ON.

Na [Figura 1](#), os círculos representam as ontologias da rede, as linhas pontilhadas representam as divisões entre as camadas (*Foundational Layer*, *Core Layer* e *Well-founded Domain Layer*). Dentre as ontologias que são apresentadas na figura, as que possuem linhas pontilhadas em torno do círculo estão em desenvolvimento, destacando-se com linhas pontilhadas na cor vermelha as ontologias diretamente utilizadas no desenvolvimento de SNOPI.

⁴ <https://nemo.inf.ufes.br/projetos/hci-on/>

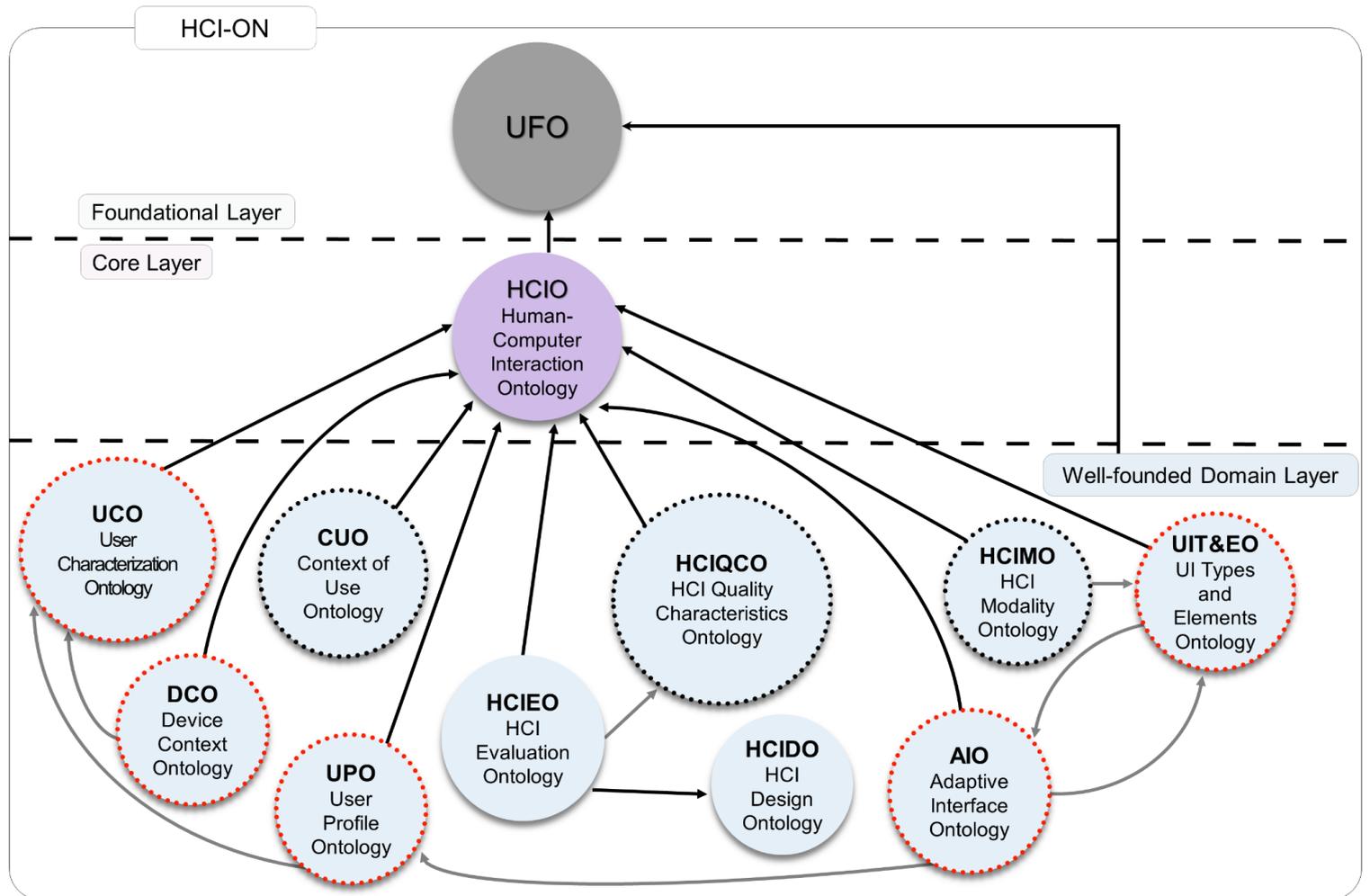


Figura 1 - Extrato de HCI-ON. Adaptado de Costa (2022).

No desenvolvimento de SNOPI (Seção 2.3.1), sistema objeto de estudo deste trabalho, foi utilizado um extrato de HCI-ON contendo conceitos das seguintes ontologias: *User Characterization Ontology* (UCO), que envolve conceitos relacionados à caracterização de usuário, *UI Types and Elements Ontology* (UIT&EO), que aborda componentes de interface com o usuário, *Adaptive Interface Ontology* (AIO), que trata das de adaptações em componentes de interface com o usuário, *Device Context Ontology* (DCO), que aborda aspectos do contexto em que se encontra o dispositivo no qual o sistema adaptativo é executado, *User Profile Ontology* (UPO), que aborda aspectos, de preferências do usuário. Também foram utilizados conceitos de SysSwO (*System and Software Ontology*), uma ontologia de SEON⁵ (*Software Engineering Ontology Network*) que possui conceitos reutilizados em HCI-ON.

⁵ <https://dev.nemo.inf.ufes.br/seon/>

Na [Figura 2](#) é apresentado o extrato da rede utilizado no desenvolvimento de SNOPI. Para simplificação, a camada *Foundational Layer* (UFO) foi omitida. As linhas pretas tracejadas separam conceitos de diferentes camadas da rede e de diferentes ontologias (o nome de cada ontologia e respectiva camada são indicados à direita). Diferentes cores são utilizadas para indicar conceitos de diferentes ontologias. Os conceitos com bordas destacadas com linhas tracejadas em vermelho foram diretamente utilizados no desenvolvimento de SNOPI, tendo sido utilizados para a implementação de uma ontologia operacional e na modelagem conceitual do sistema.

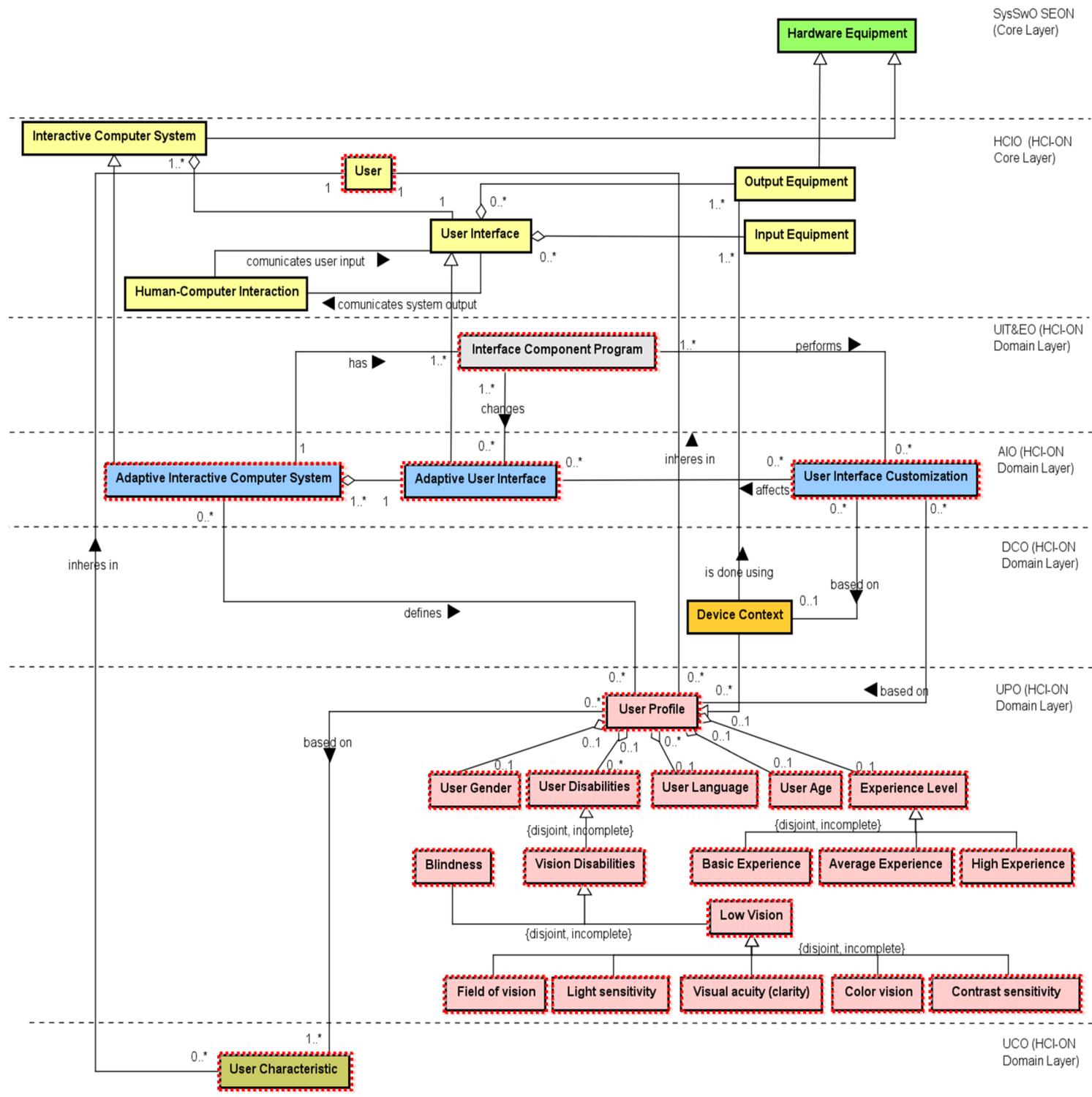


Figura 2 - Fragmento de HCI-ON utilizado no desenvolvimento de SNOPI. Adaptado de (FREITAS *et al.*, 2022).

Interactive Computer System é um sistema que possui uma *User Interface*, composta de *Output Equipment* e *Input Equipment*. Uma *Adaptive User Interface*, por sua vez, é uma *User Interface* que se adapta e, por isso, é parte de um *Adaptive Interactive Computer System*. *User* participa em *Human-Computer Interactions* para se comunicar com o sistema. *User* possui *User Characteristics* (e.g, Maria - uma usuária - tem 18 anos) que são consideradas para definir o *User Profile*, que é composto por diversas propriedades (e.g., *User Gender*, *User Disability*, *User Age*, *User Education*, *User Language*, *User Experience Level*, entre outros). Cada propriedade pode ser derivada de outras (e.g., *Cognitive Disability*, *Motor Disability*, *Auditory Disability* e *Vision Disability* são diferentes tipos de *User Disability*). *User Profiles* são definidos por *Adaptive Interactive Computer Systems*. Uma *User Interface Customization* refere-se a uma adaptação a ser feita na *Adaptive User Interface*, que é baseada no *User Profile* (o mínimo de cardinalidade desta relação é 0, pois a *User Interface Customization* pode ser baseada em outros fatores como *Device Context*). A *User Interface Customization* é feita por um *Interface Component Program* (que é um programa que compõe o *Adaptive Interactive Computer System*) e muda a *Adaptive User Interface* (FREITAS *et al.*, 2022).

Este trabalho fez uso do mesmo extrato de HCI-ON ([Figura 1](#)) utilizado para desenvolver SNOPI e de novos conceitos que foram incluídos ao longo deste trabalho para ser possível incluir novas adaptações e evoluir o sistema, aplicando a abordagem OADAPT ([Seção 2.3.2](#)).

2.3 Uso de Ontologias no Desenvolvimento de Interfaces do Usuário Adaptativas

O uso de ontologias é uma abordagem promissora para auxiliar no desenvolvimento de interfaces do usuário adaptativas. Ontologias capturam e organizam o conhecimento e, assim, podem ser usadas para estruturar conhecimento sobre o sistema interativo e as características dos usuários, auxiliando na compreensão do funcionamento de sistemas com interfaces adaptativas e servindo de base para estruturá-los, identificar as adaptações necessárias e implementá-las (FREITAS *et al.*, 2022).

Por exemplo, em (Soui *et al.* 2017), é proposta uma abordagem baseada em ontologia para sugerir adaptações de interface automaticamente, de acordo com o contexto do usuário e usando regras SWRL (*Semantic Web Rule Language*). Já em (Liu e Huang *et al.* 2016), é proposto um método baseado em ontologia para fornecer

interfaces de navegação adaptáveis e personalizáveis de acordo com a função e características do usuário. Kultsova *et al.* (2019), por sua vez, combinaram abordagens baseadas em ontologias e padrões para fornecer adaptabilidade e personalização de interfaces. Utilizando essa abordagem, os autores desenvolveram um editor de interfaces, tornando possível desenvolver interfaces adaptativas para aplicações multiplataforma assistivas.

Os trabalhos mencionados anteriormente utilizaram abordagens baseadas em ontologias para desenvolver sistemas/soluções com interface do usuário adaptativa. Porém, foi observado que apenas um deles fez reuso de ontologias existentes e nenhum utiliza ontologias de referência, que contribuem para o desenvolvimento de ontologias que se aproximam mais dos conceitos do mundo real (FREITAS *et al.*, 2022). Essas e outras limitações das abordagens existentes estão sendo exploradas na pesquisa de doutorado do coorientador deste projeto de graduação. Como mencionado anteriormente, no contexto desta pesquisa de doutorado foram desenvolvidos SNOPI e OADAPT, que serão apresentados brevemente nas próximas seções.

2.3.1 SNOPI (*Social Network with Ontology-based adaptive Interface*)

O sistema SNOPI é uma rede social voltada para a área acadêmica. Com ele, os usuários podem compartilhar postagens e seguir outros usuários para acompanhar suas publicações. Além disso, o sistema também oferece funcionalidades para que o usuário organize e publique links úteis, como trabalhos, artigos publicados e informações sobre conferências. O sistema foi desenvolvido durante a pandemia de COVID-19, quando o acesso a eventos científicos foi facilitado. Com a realização de eventos na modalidade online e com inscrições com valores reduzidos ou até mesmo gratuitas, mais pessoas tiveram a oportunidade de participar de congressos, conferências, workshops, simpósios e outros eventos científicos. Assim, SNOPI é útil para aqueles envolvidos na área acadêmica, pois permite que eles compartilhem e acessem informações relevantes sobre eventos de maneira rápida e eficiente (SCALSER, 2022).

SNOPI utilizou um fragmento de HCI-ON ([Figura 1](#)) como ontologia de referência. Com base na ontologia de referência, foi feita a modelagem conceitual do sistema e desenvolvida a ontologia operacional. A seguir, as principais funcionalidades de SNOPI são brevemente apresentadas.

Após o usuário realizar seu primeiro login no sistema (Figura 3), ele é identificado como um novo usuário e são apresentadas algumas telas com perguntas que buscam obter dados sobre ele para avaliar seu perfil. Esse questionário (Figura 4) é composto por algumas perguntas que ajudam a identificar as adaptações que serão realizadas na interface para aquele usuário. Responder à pesquisa é opcional, mas se o usuário optar por não responder, a interface não será adaptada de acordo com as suas necessidades e preferências. É importante destacar que, mesmo sem essas adaptações, o usuário ainda poderá utilizar o sistema normalmente.

Após o usuário responder à pesquisa, as suas respostas são enviadas ao servidor *back-end* do sistema. Essas informações são processadas e utilizadas para inferir um perfil de adaptações para o usuário de acordo com as suas respostas. Essa inferência é feita utilizando a ontologia operacional do sistema, que é um conjunto de regras e conceitos que permitem ao sistema adaptar a interface de acordo com as necessidades e preferências do usuário. Dessa forma, o sistema pode oferecer uma experiência de uso mais personalizada e adaptada ao perfil de cada usuário (SCALSER, 2022).

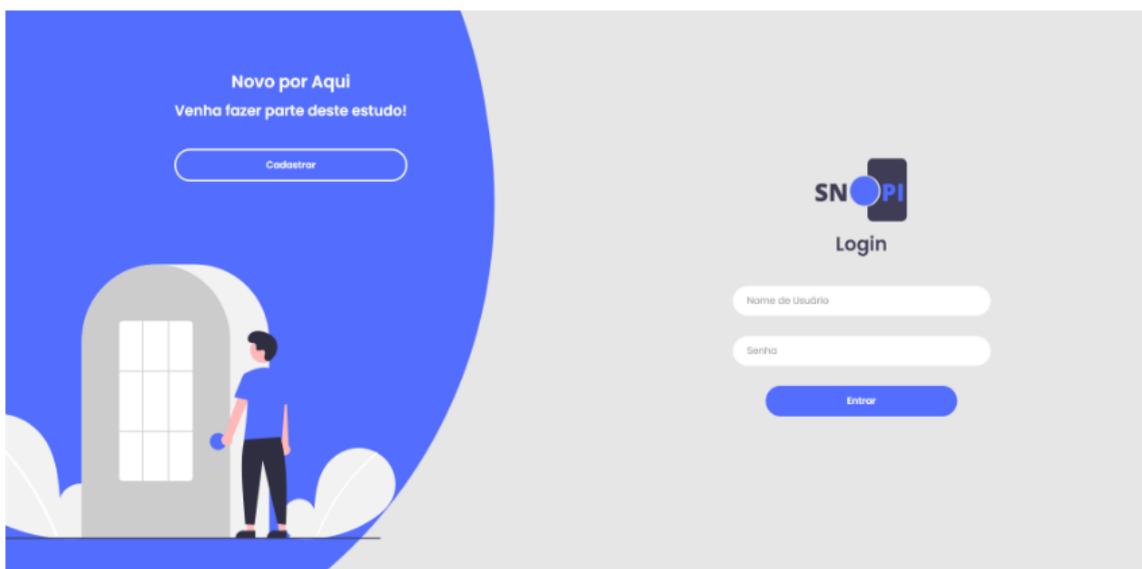


Figura 3 - Tela de Login e Cadastro de Usuário (SCALSER, 2022).

Busque aqui...

Pesquisa

Você possui alguma dificuldade ao acessar certos tipos de sites?

Sim
 Não

Você possui algum grau de Dautonismo?

Sim
 Não

Você possui algum grau de Déficit Visual?

Sim
 Não

Você possui sensibilidade a Luminosidade?

Sim
 Não

Este tamanho de texto é confortável para você?

Sim
 Não

Você possui algum problema auditivo?

Sim
 Não

Você possui algum grau de Cegueira?

Sim
 Não

Você possui algum problema com o Campo de Visão?

Sim
 Não

Você possui algum grau de Miopia?

Sim
 Não

X Rejeitar ✓ Finalizar

Figura 4 - Questionário exibido ao usuário após seu primeiro login (SCALSER, 2022).

A [Figura 5](#) ilustra a adaptação da interface do usuário com o modo escuro (baixa emissão de luz) e o aumento do tamanho da fonte. Essa adaptação é adequada para usuários daltônicos ou sensíveis à luz, bem como para usuários com deficiência visual. O modo escuro reduz a quantidade de luz emitida pela tela, o que é indicado para quem é sensível à luz ou tem dificuldade em enxergar em ambientes com muita luminosidade. Já o aumento do tamanho da fonte facilita a leitura para quem tem dificuldade de visão ou precisa de letras maiores para enxergar com mais facilidade. Dessa forma, a adaptação da interface para essas condições é uma forma de tornar o sistema mais acessível e fácil de usar.

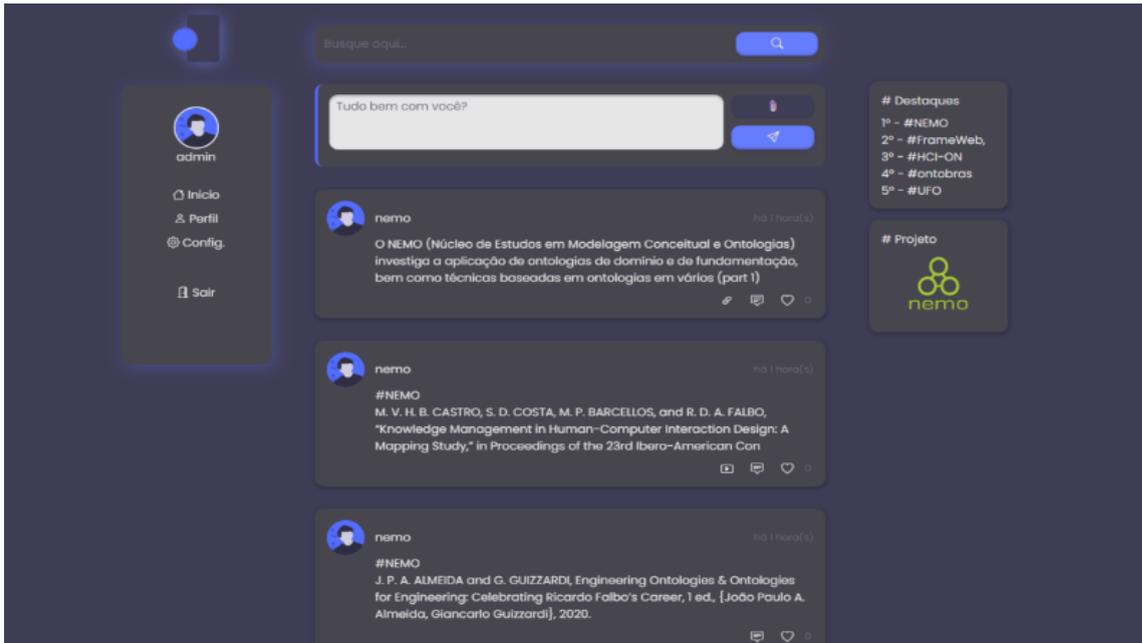


Figura 5 - Tela de Feed do sistema com adaptação - Modo escuro e fonte de letra aumentada (SCALSER, 2022).

2.3.2 OADAPT (*Ontology-based Approach to Develop AdaPtive inTerfaces*)

A abordagem baseada em ontologia para desenvolver interfaces adaptativas OADAPT consiste em (i) um *framework* de conhecimento contendo ontologias que abordam aspectos relevantes para sistemas com interface adaptativa (esse *framework* consite em um conjunto de ontologias de HCI-ON) e (ii) um processo que descreve as etapas para usá-lo para desenvolver tais sistemas (FREITAS *et al.*, 2022). O processo foi definido a partir da experiência de desenvolvimento do sistema SNOPI e considera o uso de ontologias em *dev-time* (*Ontology-driven Development* (ODD)) e *runtime* (*Ontology-based Architectures* (OBA)). Por ser um processo de desenvolvimento de software, foram incluídas algumas fases clássicas do desenvolvimento de software.

A [Figura 6](#) mostra uma visão geral do processo, que possui oito etapas para desenvolver sistemas com interfaces adaptativas, as quais são descritas em seguida.

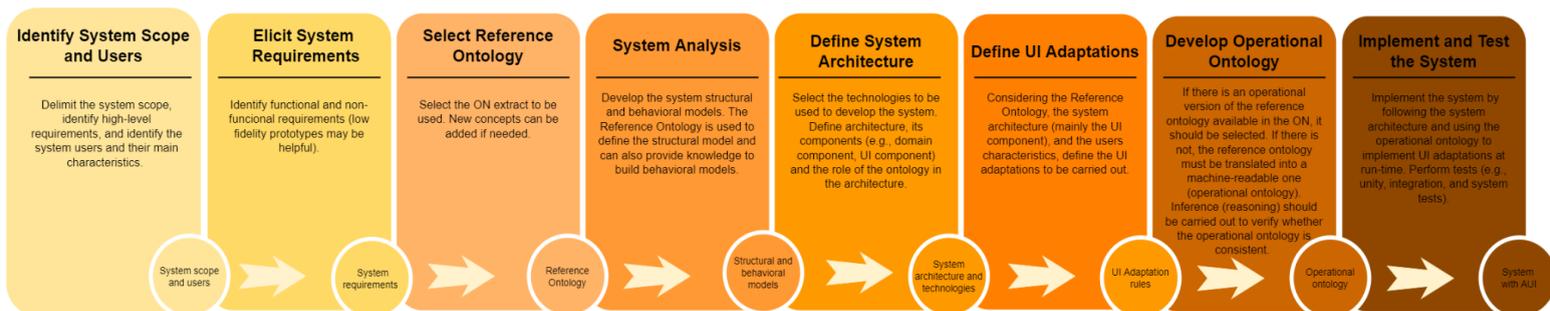


Figura 6 - Visão geral do processo baseado em ontologia para desenvolver sistemas com interfaces adaptativas (FREITAS *et al.*, 2022).

- i. **Identificar o escopo e os usuários do sistema:** consiste em delimitar o escopo do sistema, identificando os requisitos de alto nível, os usuários do sistema e suas características. Se concentra em entender o propósito e os limites do sistema, o problema a ser abordado e os diferentes tipos de usuários esperados. Conhecer as características dos usuários ajudará a definir as adaptações necessárias na interface com o usuário nas próximas etapas. As principais necessidades dos usuários também devem ser identificadas para estabelecer um conjunto inicial de funcionalidades com base em requisitos de alto nível. A adoção de técnicas como Mapa de Empatia⁶ e Personas⁷ é útil para identificar os usuários, suas necessidades e características.
- ii. **Elicitar requisitos do sistema:** nesta etapa, os resultados da etapa anterior são refinados, definindo os requisitos funcionais e não funcionais do sistema. Os requisitos funcionais descrevem as funções que o sistema deve conter. Os requisitos não funcionais, por sua vez, representam restrições que o sistema deve abordar. Nesse sentido, os requisitos não funcionais são particularmente importantes para indicar as necessidades de adaptação da interface com o usuário. Por exemplo, se os usuários do sistema precisarem de opções de acessibilidade, os requisitos não funcionais que especificam essas necessidades devem ser definidos para serem abordados na interface com o usuário. Protótipos de baixa fidelidade podem ser úteis nesta etapa para capturar informações sobre recursos e requisitos não funcionais.

⁶ Mapa de Empatia é uma ferramenta de *design thinking* que é usada para ajudar a compreender as necessidades, desejos e sentimentos de um usuário (GRAY; BROWN; MACANUFO, 2010).

⁷ *Personas* são personagens fictícios criados para representar diferentes tipos de usuários. Elas são usadas em *design thinking* como uma maneira de ajudar a compreender as necessidades e desejos dos usuários (COOPER, 1999).

- iii. **Selecionar ontologia:** nesta etapa, considerando o escopo do sistema, requisitos e características do usuário, deve-se selecionar o fragmento da rede de ontologias HCI-ON (*i.e.*, a ontologia de referência) a ser utilizado. Se necessário, novos conceitos podem ser adicionados ao fragmento selecionado. A ontologia de referência pode ser usada para refinar os resultados produzidos em (i) e (ii). Por exemplo, a ontologia pode revelar novas características do usuário a serem consideradas e refinar requisitos não funcionais.
- iv. **Realizar análise do sistema:** compreende o desenvolvimento dos modelos estruturais e comportamentais do sistema. A ontologia de referência é usada como base para o modelo estrutural (*e.g.*, diagrama de classes). Se necessário, o modelo pode ser ajustado para ser mais adequado ao sistema (*e.g.*, alguns conceitos podem ser transformados em atributos de outros conceitos).
- v. **Definir a arquitetura do sistema:** consiste em definir a arquitetura do sistema, seus componentes (*e.g.*, componente de domínio do problema, componente de interface do usuário) e tecnologias relacionadas. Ao adotar a OBA, a arquitetura deve incluir componentes que retratam o uso da ontologia operacional. O papel da ontologia operacional deve ser claramente definido (*e.g.*, seu uso no mecanismo de raciocínio semântico (*reasoning*)). Além disso, ao selecionar as tecnologias, a linguagem e as tecnologias para implementar a ontologia operacional também devem ser definidas.
- vi. **Definir adaptações de interface do usuário:** nesta etapa são definidas as regras de adaptação da interface do usuário. Assim, considerando a ontologia de referência, a arquitetura do sistema (principalmente o componente de interface do usuário) e as características do usuário, as adaptações da interface a serem realizadas devem ser definidas. Por exemplo, pode-se definir que se o usuário for sensível ao brilho, a tela deve ser colocada no modo de alto contraste. Neste estágio, pode não haver preocupação com a linguagem de máquina que irá implementar as regras. A ontologia de referência apoia o mapeamento das características do usuário para elementos específicos da interface do usuário ou necessidades de adaptação. Isso ajuda a definir as regras de adaptação. Além disso, padrões, diretrizes e padrões existentes podem ser considerados para ajudar a definir adaptações

adequadas (*e.g.*, o padrão de acessibilidade do W3C⁸ (*World Wide Web Consortium*) e as Diretrizes de Acessibilidade de Conteúdo da Web⁹) podem ser usados para ajudar a definir adaptações de acessibilidade.

vii. Desenvolver ontologia operacional: consiste em produzir a ontologia operacional que será utilizada em tempo de execução. Se houver uma versão operacional da ontologia de referência disponível na rede de ontologias, ela deve ser selecionada. Se não houver, a ontologia de referência deve ser traduzida em uma ontologia legível por máquina. *Reasoning* deve ser realizado para verificar a consistência da ontologia operacional¹⁰. As regras de adaptação definidas no passo (vi) devem ser incorporadas como axiomas na ontologia operacional. Em tempo de execução, o mecanismo de raciocínio semântico definido na arquitetura do sistema usa os axiomas para fazer inferências nos dados do usuário e identificar a adaptação da interface do usuário mais adequada para um determinado usuário. Recomenda-se criar alguns casos de teste e instanciá-los na ontologia operacional para verificar se as regras de adaptação (axiomas) são consistentes. Para essa verificação, após a população da ontologia operacional, é necessário realizar o *reasoning*. Isso pode ser feito usando ferramentas como o Protégé¹¹. A ontologia operacional (*e.g.*, o arquivo OWL) resultante desta etapa será utilizada na próxima etapa para implementar o sistema.

viii. Implementar e testar o sistema: nesta etapa, o sistema é implementado seguindo a arquitetura definida e usando a ontologia operacional para implementar as adaptações da interface do usuário em tempo de execução. Testes unitários, de integração e de sistema devem ser realizados para garantir que o sistema atende adequadamente aos requisitos funcionais e não funcionais estabelecidos. Além disso, testes de usabilidade são necessários para avaliar a interface com o usuário.

⁸ <https://www.w3.org/>

⁹ <https://www.w3c.br/traducoes/wcag/wcag21-pt-BR/>

¹⁰ Uma vantagem do uso do HCI-ON (*Human-Computer Interaction Ontology Network*) é que ela fornece as ontologias de referência e também suas versões operacionais em OWL (*Web Ontology Language*).

¹¹ <https://protege.stanford.edu/>

2.4 Tecnologias Utilizadas neste Trabalho

Nesta seção são apresentadas brevemente as tecnologias utilizadas neste trabalho, que são as mesmas utilizadas em (SCALSER, 2022). SNOPI foi desenvolvido como uma aplicação baseada na Web, mais conhecido como WebApp (PRESSMAN; LOWE, 2009), se baseando no modelo cliente-servidor, onde é denominada de *front-end* a parte do cliente e de *back-end* a parte do servidor.

O *back-end* foi implementado utilizando o *Framework Spring*¹² baseado no padrão de arquitetura *Spring Boot*¹³, que é um módulo do *Framework Spring*. O *Spring Boot* baseia-se no modelo *Model-View-Controller* (MVC), que é um padrão de arquitetura de software.

A implementação do *back-end* utilizando o Spring no SNOPI utilizou as seguintes linguagens, conforme explicado em (SCALSER, 2022):

- **JAVA**¹⁴: Linguagem de programação orientada a objetos de código aberto, sendo uma das linguagens mais utilizadas para desenvolvimento de aplicações Web e Mobile.
- **PostgreSQL**¹⁵: É um sistema de gerenciamento de banco de dados objeto relacional, que utiliza e estende a linguagem de consulta estruturada, o SQL.

Já o *front-end* da aplicação foi implementado a partir do padrão *Model-View-ViewModel* (MVVM). Este padrão foi inspirado no *MVC* (*Model-View-Controller*) e apresenta como principal diferença a camada *ViewModel*, que se beneficia do recurso de *data binding*¹⁶ (SCALSER, 2022). O *framework Angular*¹⁷ foi utilizado para o desenvolvimento de SNOPI e orienta-se pelos padrões tanto do MVC quanto do MVVM. Em SNOPI foi utilizada a versão 10.0.0 do *Angular* com compatibilidade com a ferramenta *JHipster*¹⁸. Na implementação do *front-end* foram utilizadas as seguintes linguagens, conforme explicado em (SCALSER, 2022):

¹² O *Spring Framework* é uma plataforma de desenvolvimento Java que fornece recursos para a criação de aplicações de software.

¹³ <https://docs.spring.io/spring-boot/docs/current/reference/html/>

¹⁴ <https://www.java.com/>

¹⁵ <https://www.postgresql.org/>

¹⁶ *Data binding* é uma técnica que permite a ligação de dados entre elementos de uma interface de usuário (UI) e elementos de dados.

¹⁷ <https://angular.io/>

¹⁸ <https://www.jhipster.tech/>

- **HyperText Markup Language (HTML):** Linguagem de marcação utilizada para a construção e organização de elementos da página Web.
- **TypeScript:** Linguagem que estende *JavaScript* e tem como características ser interpretada e estruturada, estaticamente tipada, multiparadigma e assíncrona. Ela é utilizada em *frameworks* para facilitar a comunicação entre componentes e aplicar *scripts*.
- **Sassy Cascading Style Sheets (SCSS¹⁹):** Linguagem que expande a sintaxe do CSS (*Cascading Style Sheets* - linguagem de estilização), utilizada para aplicar estilos nos componentes do HTML.

Para implementar as novas adaptações, foram utilizadas também, uma API nativa do Javascript e duas bibliotecas, sendo elas:

- **Web Speech API²⁰:** É uma API nativa do Javascript que contém uma Interface (*SpeechRecognition API*) para reconhecimento assíncrono de ‘discursos’. Essa interface fornece a habilidade para reconhecer o contexto de voz de uma entrada de áudio (normalmente por meio do serviço de reconhecimento de ‘discursos’ padrão do aparelho) e responde ‘apropriadamente’.
- **TensorFlow.js²¹:** É uma biblioteca JavaScript acelerada por hardware de código aberto para treinamento e deploy de modelos de *machine learning*. Fornece modelos para auxiliar na detecção em tempo real de gestos com as mãos, é utilizado em conjunto com a *library FingerPose*.
- **FingerPose²²:** É uma *library* de JavaScript que fornece um classificador para reconhecer gestos das mãos, consumindo modelos fornecidos pelo pacote *Hand-pose-detection²³* do *TensorFlow.js*.

2.5 Considerações Finais do Capítulo

Este capítulo apresentou fundamentos teóricos e informações sobre as tecnologias relevantes para o desenvolvimento deste trabalho. Com relação aos aspectos teóricos, foram abordados tópicos relacionados a interfaces adaptativas,

¹⁹ <https://sass-lang.com/>

²⁰ https://developer.mozilla.org/en-US/docs/Web/API/Web_Speech_API

²¹ <https://github.com/tensorflow/tfjs>

²² <https://www.npmjs.com/package/fingerpose>

²³ <https://github.com/tensorflow/tfjs-models/tree/master/hand-pose-detection>

ontologias, uso de ontologias no desenvolvimento de interfaces adaptativas, SNOPI (a ferramenta que foi evoluída neste trabalho) e OADAPT (processo adotado para evoluir SNOPI), fornecendo, desta forma, a base teórica para o desenvolvimento das novas adaptações e melhorias em SNOPI. Também foram apresentadas as tecnologias utilizadas neste trabalho, incluindo o *framework Spring Boot* utilizado para criação do *back-end* e o *framework Angular*, utilizado para criação do *front-end* do SNOPI. As novas adaptações na interface do SNOPI foram desenvolvidas com base na arquitetura do sistema, além de terem sido utilizadas outras tecnologias, como a *Speech Recognition API*, *TensorFlow.js* e *FingerPose*.

Capítulo 3

Aplicação da Abordagem OADAPT para Evolução do SNOPI

Neste capítulo são apresentados os passos que foram seguidos utilizando a abordagem OADAPT para implementar novas adaptações no sistema SNOPI. Na [Seção 3.1](#) são tratados aspectos sobre a identificação do escopo e os usuários do sistema. Na [Seção 3.2](#) é apresentado o levantamento de requisitos do sistema. Na [Seção 3.3](#) é apresentada a seleção da ontologia. A [Seção 3.4](#) apresenta resultados da análise do sistema. Na [Seção 3.5](#) apresenta a definição da arquitetura do sistema. Na [Seção 3.6](#) foram definidas as adaptações de interface do usuário. Na [Seção 3.7](#) é apresentado o desenvolvimento a ontologia operacional. Na [Seção 3.8](#) apresenta implementação e testes no sistema. Por fim, na [Seção 3.9](#) são feitas as considerações finais do capítulo.

3.1 Identificar o Escopo e os Usuários do Sistema

Nesta etapa, foi realizada a identificação do escopo e dos usuários do sistema. O objetivo foi delimitar o escopo do sistema, identificar os requisitos de alto nível e conhecer os usuários envolvidos, incluindo suas características. Foi dedicado tempo para compreender o propósito e os limites do sistema, assim como identificar o problema a ser abordado. Além disso, foram avaliados os diferentes tipos de usuários esperados. A análise das características dos usuários foi fundamental para definir as adaptações necessárias nas próximas etapas.

3.1.1 Identificação dos Usuários do Sistema e suas Características

Nesta atividade, foram identificados os usuários do sistema e suas características, com base nos conceitos/recomendações da W3C *Web Accessibility Initiative* (WAI)²⁴. A [Tabela 1](#) e a [Figura 7](#), descrevem em detalhes exemplos de *Persona* e respectivamente Mapa de Empatia, que foram utilizados para identificação dos possíveis usuários que irão utilizar o sistema.

²⁴ <https://www.w3.org/WAI/people-use-web/abilities-barriers/>

Tabela 1 - Persona Alex - Usuário com Lesão por Esforço Repetitivo (LER) (ABOU-ZAHRA; SINCLAIR, 2017).

Nome: Alex
Resumo: Repórter com Lesão por Esforço Repetitivo (LER)
Descrição: Alex trabalha como repórter há mais de 20 anos e desenvolveu uma lesão por esforço repetitivo que torna doloroso usar o mouse e digitar por longos períodos de tempo. Embora tenha levado pesquisa, tempo e comprometimento consideráveis para aprender, ele é capaz de trabalhar com menos dor usando as seguintes modificações em seu ambiente de trabalho: <ul style="list-style-type: none">● Uso de teclado ergonômico.● Uso de comandos de teclado sem mouse.● Software de reconhecimento de voz em seu computador e telefone celular.● Toque assistido em seu telefone celular. Alex encontra problemas quando sites e outros conteúdos online não podem ser navegados apenas por comandos de teclado. Ele frequentemente encontra formulários da web que não possuem equivalentes de teclado. Às vezes também é difícil pular conteúdo e navegar pelas seções de uma página da Web sem usar muitos comandos de teclado, o que é muito cansativo e limita o tempo que ele pode passar trabalhando confortavelmente. Ele também começou recentemente a explorar o uso do toque assistido em seu telefone celular, o que também o ajuda a usar menos gestos e trabalhar por períodos mais longos. Como uma correção temporária, o empregador de Alex criou várias soluções alternativas personalizadas que fornecem suporte de teclado para o uso do sistema de gerenciamento de conteúdo interno da empresa com a intenção de implementar suporte de teclado aprimorado para beneficiar todos os usuários no próximo lançamento do software.

A [Figura 7](#) ilustra o mapa de empatia baseado no *Person Alex* ([Tabela 1](#)), destacando as frustrações e os obstáculos enfrentados devido à Lesão por Esforços Repetitivos (LER). A figura também destaca a busca de soluções, a importância da escuta ativa, a observação de exemplos positivos e a iniciativa de comunicação e busca por padrões de acessibilidade.

Descrição do Cenário



Alex

Alex, um repórter com Lesão por Esforço Repetitivo (LER), enfrenta dificuldades para navegar em sites usando apenas comandos de teclado. A falta de suporte de teclado em formulários e a necessidade de muitos comandos de teclado limitam seu tempo de trabalho. Ele está explorando o uso do toque assistido no celular. Seu empregador criou soluções temporárias de suporte de teclado. O objetivo é melhorar a experiência online de Alex e outros usuários, permitindo a navegação também por comandos de voz e gestos da mão.

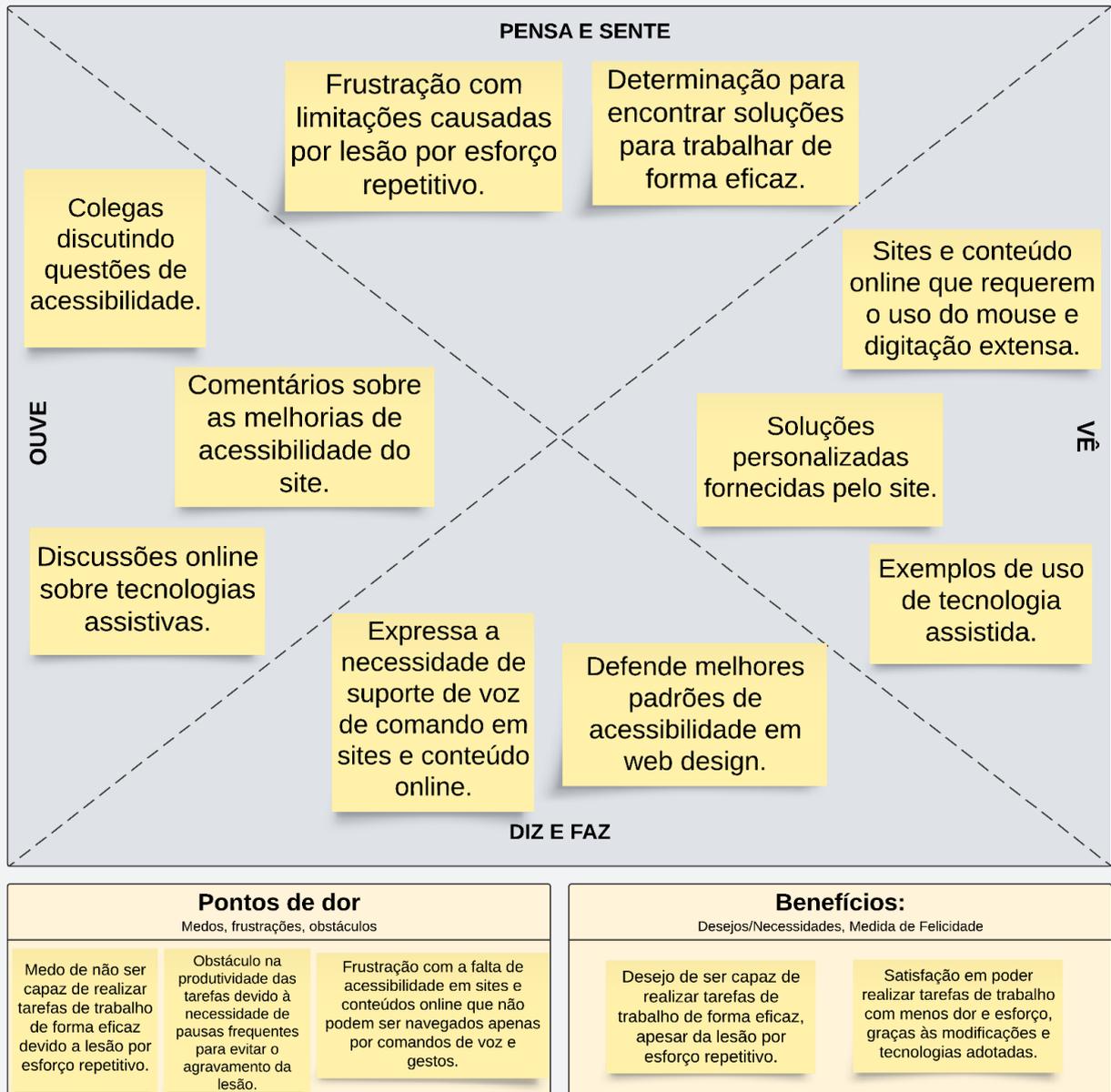


Figura 7 - Mapa de Empatia baseado na Persona Alex.

Segundo Abou-Zahra e Sinclair (2017), pessoas com **Deficiências Físicas** podem enfrentar dificuldades ao interagir com elementos na Web e têm maior propensão a cometer erros durante a digitação e clicar em elementos da interface. Exemplos de deficiência física: Amputação, Artrite, Lesão por esforço repetitivo

(LER), Quadriplegia, entre outros. Já pessoas com **Deficiências Cognitivas, de Aprendizado e Neurológicas** utilizam diferentes tipos de meios de navegação na web, dependendo de suas necessidades em particular. Alguns usuários utilizam ferramentas que redimensionam o texto e o espaçamento ou personalizam as cores para assim facilitar a leitura. Exemplos de Deficiências Cognitivas, de Aprendizado e Neurológicas: Deficiências Intelectuais, Dificuldades de Aprendizagem, entre outros. Por fim, pessoas com **Deficiências Visuais** podem variar de leve a cegueira, incluindo variações na percepção de cores e brilho, independentemente da acuidade visual. Exemplos de Deficiências Visuais: Daltonismo, Baixa visão, Cegueira, entre outros.

3.1.2 Identificação de Novas Adaptações

Nesta atividade, considerando-se as características dos usuários identificados, foram definidas as novas adaptações necessárias ao sistema. Cabe destacar que, embora OADAPT inclua mais adiante uma atividade específica para definição de adaptações da interface com o usuário a ser implementadas, é importante ter uma visão geral das adaptações desde o início, as quais podem ser refinadas mais adiante. A [Tabela 2](#) descreve as novas adaptações definidas, como também, a descrição de cada adaptação e qual o perfil de usuário é indicado para a adaptação descrita.

Tabela 2 - Novas adaptações definidas.

Adaptação	Descrição	Perfil do Usuário
Navegar no sistema por meio de gestos das mãos.	Rolar a página para cima, para baixo ou topo utilizando as mãos.	Deficiência Física ²⁵ ; Deficiências cognitivas, de aprendizagem e neurológicas ²⁶ .
Navegar no sistema por meio de Comando de voz	Executar comandos com um microfone (por exemplo: o usuário ao falar 'rolar tela para cima' a tela irá rolar para cima. Escrever texto com voz para fazer postagens, entre outros) em navegadores.	Deficiência Física; Deficiências cognitivas, de aprendizagem e neurológicas.
Mudar cores da interface de acordo com contexto do usuário.	Com base na localidade/horário do equipamento do usuário. Altera a interface automaticamente para o modo escuro.	Deficiência visual ²⁷ .

²⁵ <https://www.w3.org/WAI/people-use-web/abilities-barriers/#physical>

²⁶ <https://www.w3.org/WAI/people-use-web/abilities-barriers/#cognitive>

²⁷ <https://www.w3.org/WAI/people-use-web/abilities-barriers/#visual>

3.2 Elicitar Requisitos do Sistema

Na etapa de levantamento de requisitos é necessário coletar informações sobre as funções que o sistema deve desempenhar (requisitos funcionais) e as limitações que ele deve respeitar (requisitos não funcionais). Essas informações são obtidas por meio de entrevistas com usuários, clientes e outras partes interessadas, como também análise de sistemas e documentos relevantes (FALBO, 2017). Nesta seção, serão apresentados os requisitos identificados.

Visando atender as novas adaptações do sistema descritas anteriormente (Seção 3.1.2), foi realizada uma revisão do requisito funcional RF03 de SNOPI (Tabela 3) (*i.e.*, a interface do sistema deve se adaptar de acordo com características e preferências do usuário) e foram definidos novos requisitos não funcionais relacionados à adaptabilidade (Tabela 4).

Tabela 3 - Requisito funcional ajustado.

ID	Requisito
RF03	O sistema deve permitir armazenar informações sobre as características do usuário (considerando as informações relacionadas aos conceitos <i>'Cognitive_Disability'</i> e <i>'Physical_Disability'</i> do <i>'User_Profile'</i> apresentados na ontologia).

Tabela 4 - Novos requisitos não funcionais identificados.

ID	Requisito
RNF01	O sistema deve permitir ao usuário com os gestos pré-definidos da mão/dedos do usuário por meio da Webcam, rolar a tela com gestos das mãos/dedos.
RNF02	O sistema deve permitir que os usuários executem comandos de voz utilizando o microfone do equipamento, para navegar pelo site e fazer postagens sem usar o mouse ou teclado.
RNF03	O sistema deve diferenciar os comandos de navegação e palavras ditadas. Ou seja, diferenciar o que são comandos de navegação (<i>e.g.</i> , 'ir para o topo' e 'subir um pouco') e atribuir textos no campo <i>post</i> (<i>e.g.</i> , 'limpar texto' ou 'enviar <i>post</i> ').

3.3 Selecionar Ontologia

Nesta etapa, foi considerado o escopo do sistema, requisitos e características dos usuários. Foi selecionado o mesmo fragmento de HCI-ON utilizado no desenvolvimento do SNOPI e foram adicionados novos conceitos referentes às novas adaptações, são eles: **Environment Codition**: *Luminosity, Time of Day*; **Interaction Devices**: *Microphone, Webcam*; **Physical Disability**: *Quadriplegia, RSI (Repetitive Stress Injury), Amputation*; **Cognitive Disability**: *Intellectual Disability, Multiple Sclerosis*. A [Figura 8](#) apresenta o fragmento de HCI-ON usado. Novos conceitos estão circulados em vermelho.

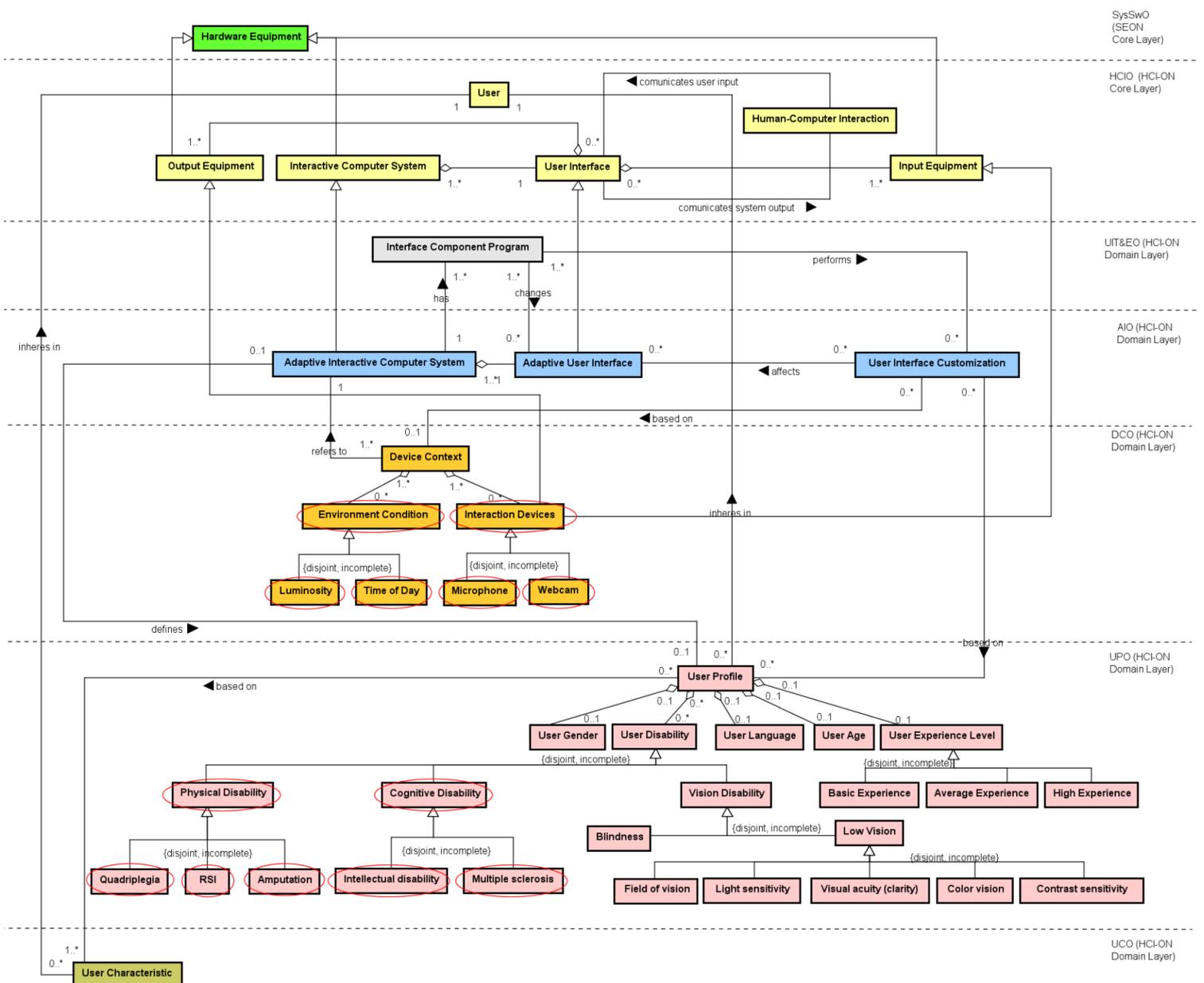


Figura 8 - Fragmento de HCI-ON incluindo novos conceitos.

3.4 Realizar Análise do Sistema

Esta etapa compreendeu o desenvolvimento dos modelos estruturais e comportamentais do sistema. A ontologia de referência ([Figura 8](#)) foi usada como base para o modelo estrutural do diagrama de classes ([Seção 3.4.1](#)).

3.4.1 Diagrama de Classes

As classes do sistema são representadas pelo modelo inicial e a modelagem de conceitual estrutural é uma atividade crucial do paradigma de desenvolvimento orientado a objetos, na qual ocorre a identificação dessas classes (FALBO, 2017). A [Figura 9](#) apresenta o diagrama de classes de SNOPI, destacando com círculo na cor vermelha as classes que foram derivadas dos novos conceitos que foram adicionados na ontologia de referência ([Figura 8](#)).

Considerando que o sistema inicial proposto é uma rede social, o modelo de classes já inclui conceitos básicos para as ações principais que uma rede social suporta, sendo algumas delas: criar postagens, curtir postagens e comentar postagens. Foram adicionados novos conceitos para adicionar/implementar novas adaptações ao sistema SNOPI2.0.

A classe **Profile** é baseada no conceito *User Profile* da ontologia de referência e armazena informações sobre as características dos usuários registrados na classe **User**. Além disso, a classe **Profile** são armazenados metadados úteis para o gerenciamento e melhoria das adaptações da interface. A classe **Sharing** tem como objetivo armazenar informações relevantes sobre os compartilhamentos de postagens. A classe **Preference** armazena as preferências do usuário em relação à interface do sistema e define como deve ser a interface para esse usuário, com base no conceito *User Interface Customization* da ontologia de referência. As preferências são definidas automaticamente pelo sistema com base nas informações contidas em **Profile** e nas inferências feitas usando a ontologia operacional. A classe **Post** é responsável por armazenar as mensagens postadas na rede social e as relações entre diferentes mensagens, que representam as interações entre os usuários do sistema (SCALSER, 2022).

Foram adicionadas novas classes para descrever e implementar as novas adaptações da interface. A classe **DeviceContext** armazena informações sobre os dispositivos do usuário, como webcam e microfone. As classes **VoiceCommandMode** e **HandsFreeMode** armazenam dados sobre a ativação ou

desativação das adaptações: navegação e criar postagens por comandos de voz e navegação por gestos das mãos/dedos, respectivamente. Já na classe **Profile**, foram adicionados novos conceitos, necessários para recomendar as novas adaptações de acordo com as dificuldades/características dos usuários, são eles: **movementLimitation**, **CognitiveDifficulty**, **LER**, **mobilityLimitation**. Os nomes em vermelho, equivalem aos nomes que foram representados na ontologia operacional ([Figura 11](#)).

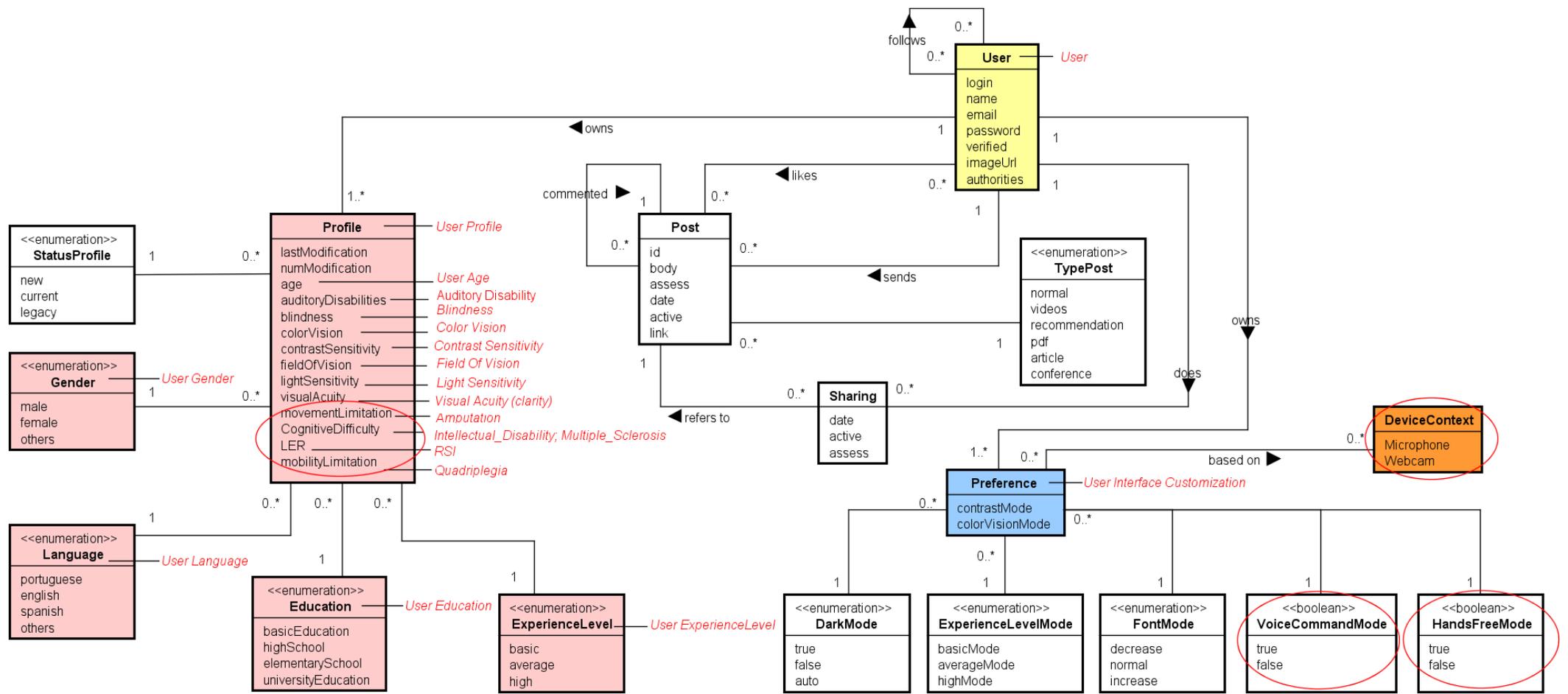


Figura 9 - Diagrama de Classes de SNOPI2.0.

3.5 Definir a Arquitetura do Sistema

Esta etapa compreendeu a definição da arquitetura do sistema. A arquitetura utilizada na implementação das novas adaptações foi a mesma utilizada em SNOPI, conforme ilustrado na [Figura 10](#).

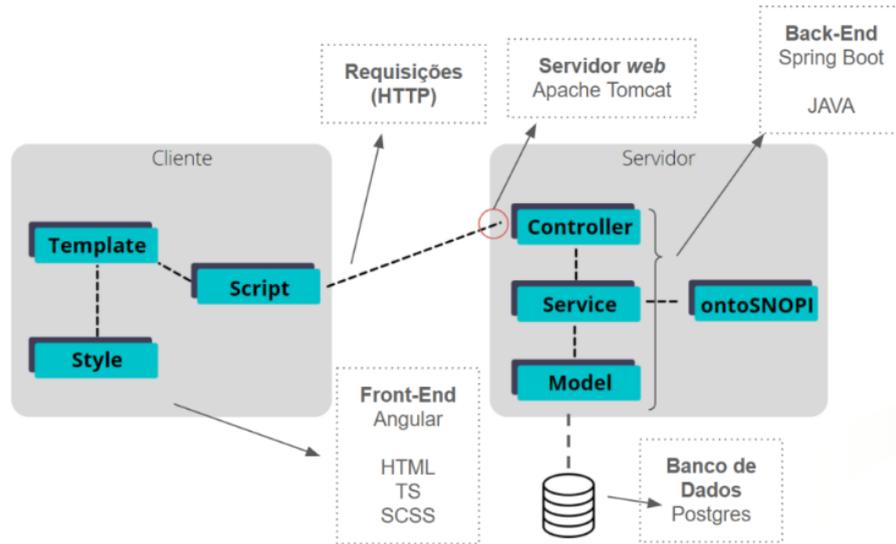


Figura 10 - Diagrama da arquitetura de software do SNOPI (SCALSER, 2022).

SNOPI foi desenvolvido seguindo o paradigma cliente-servidor, conforme ilustrado na [Figura 10](#), e, portanto, foi dividido em dois subsistemas: um executado no lado do cliente (*front-end*) e outro no lado do servidor (*back-end*). O *back-end*, que compreende o **Servidor**, foi desenvolvido utilizando o *Framework Spring*, baseado no padrão *Spring Boot*. Ele é composto pela camada de controle, camada de lógica de negócio e de gerência de dados, que correspondem, respectivamente, às camadas de *Controller*, *Service* e *Model*. O *front-end*, que compreende o **Cliente**, foi desenvolvido utilizando o *Framework Angular*, baseado no padrão MVVM. Ele é composto pela camada de interface com o usuário, responsável pelos componentes de exibição de conteúdo, e está relacionada com as camadas *View* e *View Model* do padrão MVVM.

As novas adaptações na interface com o usuário foram implementadas utilizando a arquitetura do SNOPI e outras tecnologias (*Speech Recognition API*, *TensorFlow.js* e *FingerPose*) abordadas na [Seção 2.4](#).

3.6 Definir Adaptações de Interface do Usuário

Nesta etapa, foram definidas as regras de adaptação da interface. Foram consideradas a ontologia de referência, a arquitetura do sistema (principalmente o componente de interface do usuário), os requisitos e as características do usuário. A

ontologia de referência foi fundamental para o mapeamento das características do usuário para elementos específicos da interface do sistema e/ou necessidades de adaptação. Além disso, foram levados em consideração conceitos e recomendações da W3C *Web Accessibility Initiative* (WAI) para garantir a acessibilidade do sistema.

Os novos usuários integrados ao sistema, conforme explicado na [Seção 3.1.1](#), foram: **usuários com deficiências cognitivas, de aprendizagem e neurológicas e usuários com deficiência física.**

Com base nas diretrizes da WAI, foram definidas quais novas deficiências e adaptações seriam incorporadas ao sistema, como também para quais usuários essas novas adaptações seriam indicadas conforme mostrado na [Tabela 2](#). Para capturar essas informações do usuário, foram criadas novas perguntas ([Tabela 5](#)) e adicionadas ao questionário do usuário ([Figura 21](#)). Com base nas características do usuário, foram definidas quais adaptações poderiam ser recomendadas para ele. A [Tabela 5](#) mostra as adaptações que são ativadas caso a resposta do usuário no questionário seja "sim" para as perguntas indicadas.

Tabela 5 - Novas perguntas incluídas no questionário do usuário.

Pergunta	Adaptação Ativada
Você possui LER (Lesão por Esforço Repetitivo)?	Ativa <u>Navegação por Gestos</u> (<i>Hands_Free_Mode</i>) e <u>Comando de Voz</u> (<i>Voice_Command_Mode</i>).
Você possui alguma limitação ou condição que afete seus membros superiores (por exemplo, braços, mãos)?	
Você tem alguma condição médica ou deficiência que afete a mobilidade ou função dos seus membros, como quadriplegia? A quadriplegia é uma condição que causa paralisia parcial ou total (incluindo controle motor e sensação) dos quatro membros do corpo (pernas e braços) e do tronco.	Ativa <u>Comando de Voz</u> (<i>Voice_Command_Mode</i>).
Você tem alguma condição ou dificuldade que afete a sua capacidade cognitiva? Isso pode incluir deficiências intelectuais, Esclerose Múltipla (EM) ou outras condições similares.	Ativa <u>Navegação por Gestos</u> (<i>Hands_Free_Mode</i>), <u>Comando de Voz</u> (<i>Voice_Command_Mode</i>), Interface com Ícones (<i>Basic_Experience_Mode</i>) e Aumentar tamanho da fonte (<i>Font_Increase</i>).

Na [Tabela 5](#) as novas adaptações (*i.e.*, as que foram implementadas neste trabalho) foram destacadas em texto sublinhado. Adaptações que já estavam disponíveis em SNOPI foram utilizadas, pois a W3C também recomenda essas

adaptações para os usuários considerados neste trabalho. As adaptações já existentes que também foram utilizadas neste trabalho são: Interface com Ícones (*Basic_Experience_Mode*) e Aumentar tamanho da fonte (*Font_Increase*).

3.7 Desenvolver Ontologia Operacional

Uma ontologia operacional é uma versão implementada da ontologia de referência, que é criada para ser interpretada por máquinas e tem como objetivo garantir certas propriedades computacionais desejáveis (GUIZZARDI, 2007). Nesta etapa, foram adicionados os novos conceitos e as regras na ontologia ontoSNOPI²⁸, que é a versão operacional do fragmento de HCI-ON usado para desenvolver SNOPI. A nova versão foi nomeada ontoSNOPI2.0²⁹.

3.7.1 Evolução da OntoSNOPI2.0

Durante o desenvolvimento da ontologia operacional ontoSNOPI, as classes mais gerais foram criadas primeiro, seguidas pelas classes mais específicas (SCALSER, 2022). Como ilustrado na [Figura 11](#), duas classes com conceitos mais gerais foram elaboradas: *Customization* e *User_Profile*, ambas sendo subclasses da classe nativa *owl:Thing*. A classe *Customization* contém subclasses que representam as adaptações feitas na interface do sistema, tais como: *Average_Experience_Mode*, *Basic_Experience_Mode*, *Dark_Mode*, *Desktop_Mode*, *Font_Decrease*, *Font_Increase*, *High_Experience_Mode*, *Light_Mode* e *Mobile_Mode*. A classe *User_Profile*, por sua vez, possui subclasses que representam as características, dificuldades e informações gerais sobre os usuários, tais como: *User_Age*, *User_Disability*, *Auditory_Disability*, *Vision_Disability*, *Blindness*, *Low_Vision*, *Color_Vision*, *Contrast_Sensitivity*, *Field_Of_Vision*, *Light_Sensitivity*, *Visual_Acuity*, *User_Education*, *User_Experience_Level*, *Average_Experience*, *Basic_Experience*, *High_Experience*, *User_Gender*, *User_Language* e *Use_Preferences_Profile*.

Foram adicionadas à ontoSNOPI novas classes e regras ([Figura 11](#)), utilizando os novos conceitos adicionados à ontologia de referência ([Figura 8](#)), para adequar a ontologia operacional às novas adaptações previstas para SNOPI2.0. As novas classes e subclasses que foram adicionadas (*Hands_Free_Mode*, *Voice_Command_Mode*; ***Device_Context*** (*Environment_Condition: Luminosity, Time_of_Day*; ***Interaction_Devices***: *Microphone, Webcam*; ***Cognitive_Disability***: *Intellectual_Disability*,

²⁸ <https://dev.nemo.inf.ufes.br/hcion/ontoSNOPI.owl>

²⁹ <https://dev.nemo.inf.ufes.br/snopi/ontoSNOPI2.0.owl>

Multiple Sclerosis; Physical Disability: Amputation, RSI, Quadriplegia) estão destacadas na [Figura 11](#) com círculos na cor vermelha.

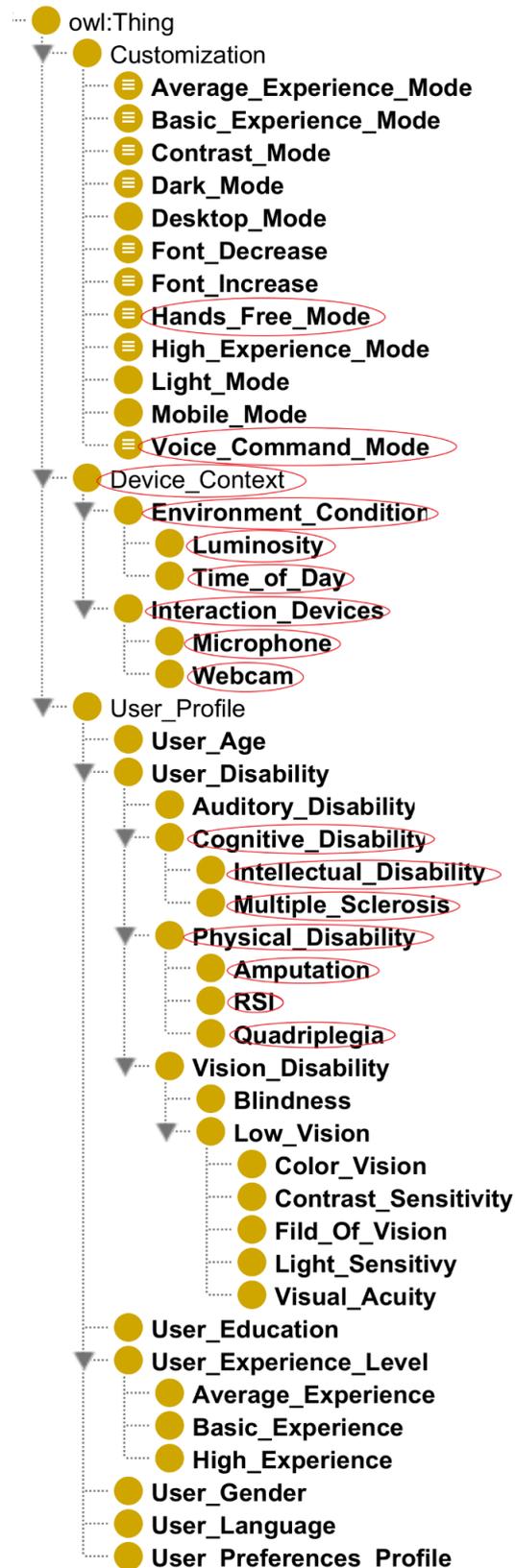


Figura 11 - Ontologia Operacional - hierarquia de classes definida.

Após a criação da ontologia no Protégé, o arquivo OWL correspondente foi gerado. Na [Tabela 6](#) são descritas as novas classes e subclasses que foram acrescentadas na ontologia operacional e suas definições.

Tabela 6 - Classes de Subclasses adicionadas a ontologia ontoSNOPI2.0.

Classe	Descrição
<i>Device_Context</i>	Representa características do contexto em que se encontra o dispositivo no qual o usuário está utilizando SNOPI (<i>e.g.</i> , características do dispositivos e do ambiente)
Subclasses Adicionadas em <i>Customization</i>	
<i>Hands_Free_Mode</i>	Indica uma adaptação de interface com o usuário para navegação utilizando gestos das mãos/dedos.
<i>Voice_Command_Mode</i>	Indica uma adaptação de interface com o usuário para navegação/postagens por meio de comandos de voz.
Subclasses Adicionadas em <i>Device_Context</i>	
<i>Environment_Condition</i>	Representa o contexto do dispositivo. Pode incluir informações sobre o ambiente físico em que o dispositivo está sendo utilizado, como localização geográfica, condições ambientais, entre outros fatores que possam afetar a interação do usuário com o dispositivo.
<i>Luminosity</i>	Indica a luminosidade do ambiente em que o usuário encontra-se.
<i>Time_of_Day</i>	Indica a hora do dia em que o usuário está utilizando o sistema
<i>Interaction_Devices</i>	Representa os dispositivos de interação, <i>i.e.</i> , dispositivos de entrada ou saída que permitem aos usuários interagir com o sistema.
<i>Microphone</i>	Dispositivo de entrada de áudio.
<i>Webcam</i>	Dispositivo capaz de capturar vídeo em tempo real e transmitir imagens, sendo utilizado como uma entrada de vídeo para diferentes tipos de aplicativos.
Subclasses Adicionadas em <i>User_Profile</i>	
<i>Cognitive_Disability</i>	Indica uma deficiência cognitiva do usuário.

Tabela 6 - Classes de Subclasses adicionadas a ontologia ontoSNOPI2.0 (continuação).

<i>Intellectual_Disability</i>	Indica deficiências intelectuais do usuário. Envolve deficiências de inteligência, aprendizado mais lento ou dificuldade de entender conceitos complexos. A síndrome de Down é uma entre muitas causas diferentes de deficiência intelectual (ABOU-ZAHRA; SINCLAIR, 2017).
<i>Multiple_Sclerosis</i>	Representa a esclerose múltipla, doença que pode causar danos às células nervosas do cérebro e da medula espinhal, afetando as habilidades auditivas, cognitivas, físicas ou visuais do usuário (ABOU-ZAHRA; SINCLAIR, 2017).
<i>Physical_Disability</i>	Representa as deficiências físicas do usuário (às vezes chamadas de deficiências motoras). Pode incluir fraqueza e limitações de controle muscular, limitações de sensação, distúrbios articulares (como artrite), dor que impede o movimento, e membros ausentes. (ABOU-ZAHRA; SINCLAIR, 2017).
<i>Amputation</i>	Indica a falta de dedos, membros ou outras partes do corpo humano (ABOU-ZAHRA; SINCLAIR, 2017).
<i>RST</i>	Indica Lesão por Esforço Repetitivo. Envolve lesões no sistema músculo-esquelético (ossos, articulações, tendões e outros tecidos conjuntivos) e no sistema nervoso de tarefas repetitivas e danos (ABOU-ZAHRA; SINCLAIR, 2017).
<i>Quadriplegia</i>	Indica (às vezes chamada de “tetraplegia”) paralisia parcial ou total (inclui controle motor e sensação) em todos os quatro membros do corpo (pernas e braços) e no tronco (ABOU-ZAHRA; SINCLAIR, 2017).

A [Figura 12](#) apresenta o axioma que estabelece as condições necessárias para que a adaptação que ativa o Modo Mãos Livres da interface com o usuário seja selecionada. Para que um indivíduo seja atribuído à classe *Hand_Free_Mode*, é necessário que o indivíduo tenha alguma das seguintes predisposições: (*has_Disabilities*): Deficiência Intelectual (*Intellectual_Disability*), Esclerose Múltipla (*Multiple_Sclerosis*) ou Lesão por Esforço Repetitivo (*RST*).

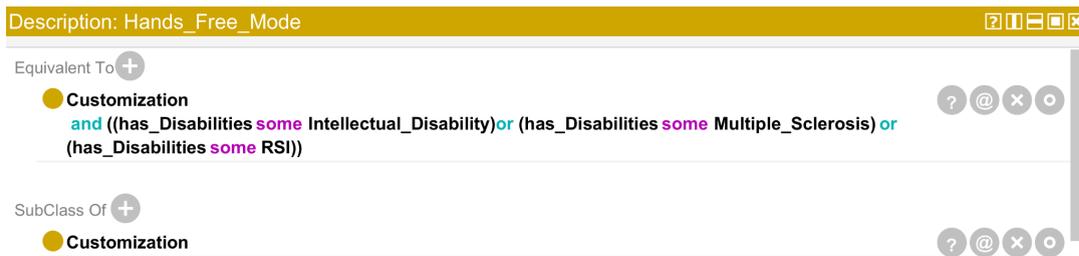


Figura 12 - Axioma da Classe Hand_Free_Mode

Por fim, a [Figura 13](#) apresenta o axioma que estabelece as condições necessárias para que a adaptação que ativa o Modo Comandos de Voz da interface com o usuário seja selecionada. Para que um indivíduo seja atribuído à classe *Voice_Command_Mode*, é necessário que o indivíduo tenha alguma das seguintes predisposições: (*has_Disabilities*): Incapacidade/Amputação dos Membros Superiores (*Amputation*), Deficiência Intelectual (*Intellectual_Disability*), Quadriplegia (*Quadriplegia*) ou Lesão por Esforço Repetitivo (*RSI*).

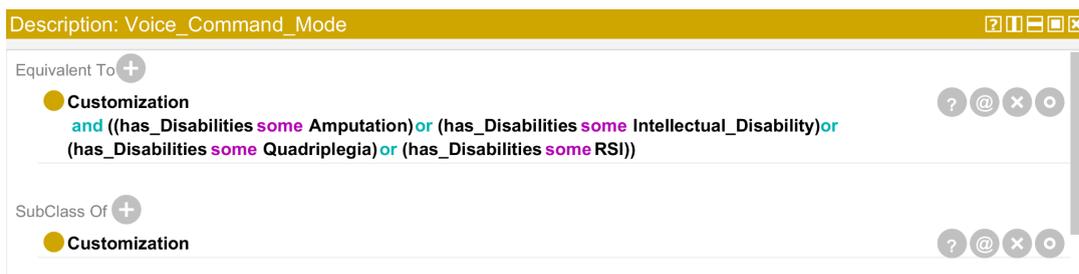


Figura 13 - Axioma da Classe Voice_Command_Mode

Para a criação dos axiomas que estabelecem as condições para que se apliquem a cada adaptação de interface com o usuário, foram considerados conceitos/recomendações da W3C *Web Accessibility Initiative* (WAI)³⁰, que explora a grande diversidade de pessoas e habilidades, destacando algumas das barreiras de acessibilidade na web que as pessoas geralmente enfrentam devido à falta de acessibilidade de sites e ferramentas da web.

Com o objetivo de avaliar a consistência da ontologia, foram realizados testes de inferência no Protégé. Esse mecanismo tem como finalidade identificar as conexões explícitas e implícitas entre as classes, para identificar possíveis inconsistências. Como exemplo, na [Figura 14](#), indica-se um indivíduo '01', que tem 34 anos e Lesão por Esforço Repetitivo (RSI), para testes de inferência. Após a inferência, foi possível

³⁰ <https://www.w3.org/WAI/people-use-web/abilities-barriers/>

identificar, como mostra a [Figura 15](#), que as adaptações de interface com o usuário para esse indivíduo devem ser “*Hands_Free_Mode*” e “*Voice_Command_Mode*”. O resultado da inferência está em conformidade com as diretrizes utilizadas como base para a elaboração dos axiomas, indicando que o teste de inferência não apresentou inconsistências.

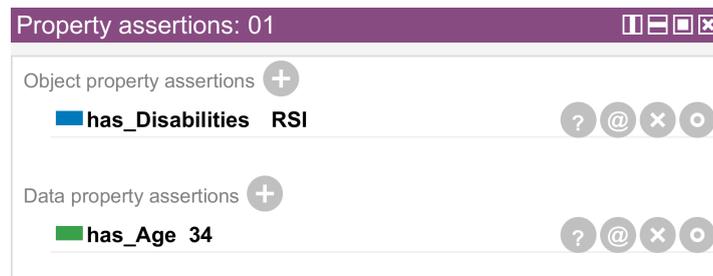


Figura 14 - Propriedades do indivíduo '01'.

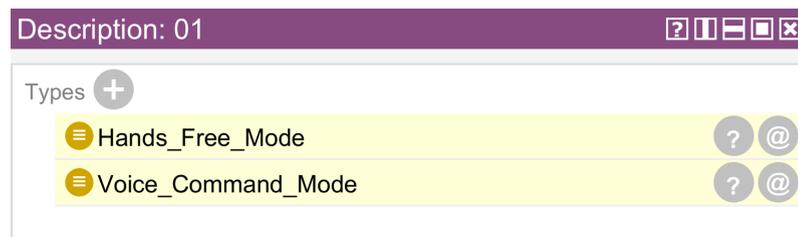


Figura 15 - Inferência sobre o indivíduo '01'.

Ao final, foi possível obter a estrutura final da ontologia utilizada no sistema, a qual está ilustrada na [Figura 16](#).

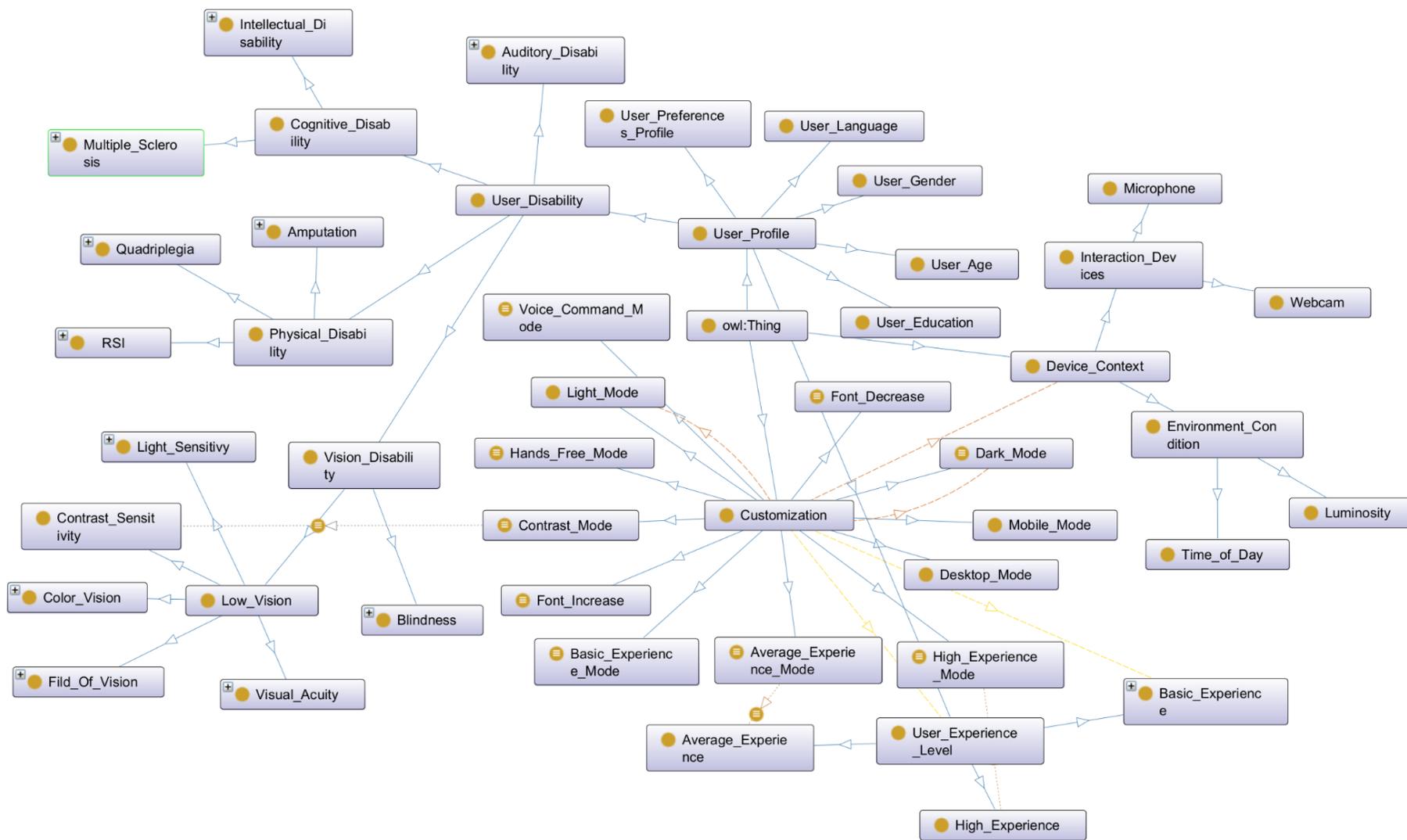


Figura 16 - Representação gráfica da ontologia operacional.

Inicialmente, também foi idealizada para ser tratada neste trabalho a adaptação *Dynamic Mode*, que consistia em adaptar automaticamente a interface com o usuário para o modo escuro (*Dark_Mode*) com base na localização, horário e luminosidade do equipamento do usuário. Foi, então, feita a adição da classe **Environment Condition** à ontologia. Essa classe contém conceitos relacionados ao horário do equipamento do usuário (*Time_of_Day*) e à luminosidade do ambiente (*Luminosity*). No entanto, devido a questões técnicas relacionadas à fidelidade da adaptação e à incorporação dos novos conceitos nas regras da ontologia operacional e, código do sistema, houveram situações não esperadas ao tentar salvar dados nas entidades do banco. Durante a implementação, não foi possível identificar a raiz do problema, o que impediu a implementação dessa adaptação no sistema. Em vez disso, foi adicionado ao sistema a opção de configuração do perfil do usuário, que permite ao usuário selecionar se ele deseja ativar, desativar ou deixar automático o modo escuro (*Dark_Mode*) do sistema.

3.7.2 Manipulação e Reasoning da OntoSNOPI2.0

Durante a execução do sistema são feitas inferências para determinar as adaptações de interface com o usuário adequadas para cada usuário. Para isso, o primeiro passo é carregar o arquivo OWL de ontoSNOPI2.0 no *back-end* do sistema, onde ocorre o processo de *reasoning*. Após o carregamento, é necessário criar uma instância da ontologia no sistema. Para isso, foi utilizada a OWL API para criar um objeto OWL Ontology, que representa a ontologia no ambiente JAVA com a OWL API.

Ao realizar o login no sistema pela primeira vez ou ao alterar seus dados na página de configurações do perfil, as informações do usuário são carregadas e armazenadas no *back-end* do sistema ([Figura 17](#)). Isso cria um indivíduo na ontoSNOPI2.0 que representa o usuário.

```
IRI ontologyIRI = IRI.create(
    "http://www.semanticweb.org/interface-adaptativa"
);
OWLDataFactory factory = manager.getOWLDataFactory();

//Carregando Individuo para inferencia
this.loadingIndividual(factory, ontologyIRI, ontology, manager, idUser);

// Reasoner
ReasonerFactory reasonerFactory = new ReasonerFactory();
Configuration configuration = new Configuration();
configuration.throwInconsistentOntologyException = false;
OWLReasoner reasoner = reasonerFactory.createReasoner(
    ontology,
    configuration
);
```

Figura 17 - Fragmento do código utilizado para manipulação da ontologia operacional (SCALSER, 2022).

Depois que o indivíduo é criado em uma instância da ontoSNOPI2.0, o Hermit Reasoner³¹ é utilizado para realizar a inferência em tempo de execução. Com base no resultado da inferência, as adaptações são feitas e os dados referentes às adaptações identificadas são armazenados no sistema. Isso garante que a interface com o usuário esteja sempre adaptada para o usuário quando ele utilizar o sistema. Uma nova adaptação na interface com o usuário só é realizada se o usuário solicitar, alterando as configurações no seu perfil.

3.8 Implementar e Testar o Sistema

Nesta etapa, foram implementadas as novas adaptações e foram realizados testes para verificar a consistência do sistema.

Com o objetivo de separar as novas implementações em um padrão de módulos, foi utilizado na implementação do *front-end* exclusivamente HTML, CSS e *JavaScript*, além de APIs nativas do *JavaScript*, para maior compatibilidade com os navegadores. Algumas APIs utilizadas, como a de Comandos de Voz, por exemplo, não são suportadas por todos os navegadores, o que pode causar erros. Sendo necessário impedir a tentativa de utilização dessas APIs em navegadores que não as suportam. Para uma melhor experiência, é recomendado utilizar o navegador Google Chrome ou navegadores que utilizam como base o Chromium³² (*e.g.*, navegador Opera).

A [Figura 18](#), mostra um fragmento de código do módulo de navegação por gestos. O módulo de Navegação por Gestos é inicializado de maneira independente dos outros módulos, assim como os demais módulos. No entanto, é possível utilizá-lo simultaneamente com o módulo de Comando de Voz, pois são separados. É importante ressaltar que há uma prioridade entre os dois módulos, a fim de evitar concorrência em ações opostas.

O módulo apresenta dois pontos importantes, a saber:

- Identificação dos gestos realizados pelo usuário com as mãos/dedos.
- Execução das funcionalidades correspondentes aos gestos identificados.

³¹ <http://www.hermit-reasoner.com/>

³² O Chromium é um projeto de navegador web de código aberto desenvolvido pela Google, que serve como base para o código-fonte do Google Chrome. Os navegadores compartilham grande parte do código-fonte e recursos.

Já a [Figura 19](#) mostra um fragmento de código do módulo de navegação/postagens por comandos de voz. O módulo possui duas ‘instâncias’ de microfones diferentes. Cada uma tem seu próprio inicializador e *#id* no HTML, desta forma há como diferenciá-los na implementação e fazê-los funcionarem independentemente. Entretanto, não é possível usá-los de maneira simultânea, para deixar um escopo de funcionalidades melhor definido para cada microfone se optou por separar suas funcionalidades e proporcionar maior manutenibilidade e legibilidade para o código.

```
const initGesturesModule = () => {
  // Coloca a div do video visível
  videoContainer = document.getElementsByClassName('video-container-sticky')[0];
  videoContainer.style.display = 'block';

  // Pega os Nodos com ids correspondentes no HTML e atribui o contexto 2D ao canvas
  video = document.querySelector("#pose-video");
  canvas = document.querySelector("#pose-canvas");
  ctx = canvas.getContext("2d");

  camBtn = document.getElementById('navigation-cam');
  camBtn.style.display = 'block';

  // Troca os ícones do botão da câmera
  camBtn.children[0].classList.remove('cam');
  camBtn.children[0].classList.add('cam');
  camBtn.classList.add('navigation-btn-blink');

  initCamera(
    VIDEO_IFRAME_CONFIG.video.width, VIDEO_IFRAME_CONFIG.video.height, VIDEO_IFRAME_CONFIG.video.fps, video
  ).then(videoStream => {
    videoStream.play();
    videoStream.addEventListener("loadeddata", () => {
      console.log("Camera is ready"); 'Camera is ready'
      main(videoStream, ctx);
    });
  }).catch(error => {
    console.log('Sistema não possui câmera para utilizar do módulo de Gestos. ');
    camBtn.style.display = 'none';
    camBtn.children[0].classList.add('cam');
    camBtn.children[0].classList.remove('cam');
    camBtn.classList.remove('navigation-btn-blink');
    videoContainer.style.display = 'none';
  });

  canvas.width = VIDEO_IFRAME_CONFIG.video.width;
  canvas.height = VIDEO_IFRAME_CONFIG.video.height;
  console.log("Canvas initialized"); 'Canvas initialized'

  // Verifica se o camBtn já tem listener de click atribuido
  if(camBtn.getAttribute('listener') !== true) {
    camBtn.addEventListener('click', () => {
      if (!video.srcObject) {
        location.reload();

        return;
      }

      disableCamera();
    });
  }
}
```

Figura 18 - Fragmento do código do módulo de navegação por gestos (Mãos Livres).

```

function initDictation() {
  // Inicializa o microfone
  recognition = initRecognition(recognition);

  // Sempre que a API de Speech achar correspondência do áudio reconhecido
  // verifica o tamanho da string gerada (event.results), manda para identificar
  // e executa o comando identificado
  recognition.onresult = function(event) {
    if (event.results.length > 0) {
      const current = event.results[event.results.length - 1][0];
      const result = current.transcript;
      const command = generalCommands(result);

      triggerCommands(command, mic);
    }
  }
};

const initMicModule = () => {
  // Verifica se o usuário possui microfone
  navigator.mediaDevices.getUserMedia({audio: true})
    .then(() => {
      initDictation();

      // Pega os Nodos com ids correspondentes no HTML e atribui listener de click no botão do microfone
      mic = document.getElementById('navigation-mic');
      mic.style.display = 'block';

      mic.addEventListener('click', function () {
        if (!recognition) {
          console.log("speech recognition API not available");
          return;
        }

        try {
          recognition.start();
        } catch (error) {
          recognition.stop();
        }
        finally {
          changeMicIcon(mic);
        }
      });

      initPostSpeech();
    })
    .catch((error => {
      // remove o display dos botões de display
      input?.style.display = 'none';
      micInside?.style.display = 'none';
      console.log('Sistema não possui microfone para utilizar do módulo de Voz.');
```

Figura 19 - Fragmento do código do módulo de navegação/postagens por comandos de voz.

3.9 Considerações Finais do Capítulo

Este capítulo apresentou as etapas da abordagem OADAPT que foram seguidas para apoiar a implementação de novas adaptações no sistema SNOPI. Foram apresentados em detalhes os resultados produzidos em cada etapa, a saber: (i) Identificar o escopo e os usuários do sistema; (ii) Elicitar requisitos do sistema; (iii) Selecionar ontologia; (iv) Realizar análise do sistema; (v) Definir a arquitetura do sistema; (vi) Definir adaptações de interface do usuário; (vii) Desenvolver ontologia operacional; e (viii) Implementar e testar o sistema. No próximo capítulo são apresentadas as novas adaptações implementadas.

Capítulo 4

SNOPI 2.0

Neste capítulo é apresentada a nova versão de SNOPI. Na [Seção 4.1](#) são apresentadas as novas adaptações implementadas. Na [Seção 4.2](#) é apresentada a adaptação de navegação no sistema por gestos das mãos. Na [Seção 4.3](#) é apresentada a adaptação de navegação/postar no sistema por comandos de voz. Por fim, na [Seção 4.4](#) são feitas as considerações finais do capítulo.

4.1 SNOPI 2.0

Nesta seção são exibidas as telas com as novas adaptações de SNOPI2.0³³, juntamente com uma breve descrição de utilização. As figuras apresentadas nesta seção ilustram as novas adaptações implementadas.

Após o usuário realizar o cadastro e realizar o primeiro *login*, o sistema exibe algumas telas com perguntas ([Figura 4](#) e [Figura 21](#)) para coletar informações e avaliar o perfil do usuário. Foram adicionadas novas perguntas ([Figura 21](#)) para capturar as informações necessárias para identificar as adaptações de interface com o usuário adequadas para os novos usuários incorporados ao sistema. É importante destacar que, embora o questionário seja opcional, se o usuário optar por não respondê-lo, a interface com o usuário não será adaptada de acordo com suas necessidades e preferências. Portanto, responder ao questionário pode melhorar a experiência do usuário ao utilizar o sistema. As respostas são enviadas ao servidor *back-end*, que utiliza a ontologia operacional para inferir um perfil de adaptações. As adaptações são aplicadas na interface do usuário após a atualização automática do sistema.

Os comandos possíveis das adaptações estão listados na [Figura 20](#) e na [Tabela 7](#) com suas respectivas funções, onde é possível ter uma visão geral sobre os comandos aceitos pelo sistema. Foi realizado um teste no sistema, que simulou a experiência do *Persona Alex* ([Tabela 1](#)), e todo o processo foi gravado em vídeo³⁴.

³³ <https://dev.nemo.inf.ufes.br/snopi/2.0/>

³⁴ https://dev.nemo.inf.ufes.br/snopi/2.0/Using_SNOPI2.0.mp4

Como utilizar as adaptações

Neste documento apresentamos todos os comandos possíveis para utilização na plataforma, seja por gesto ou por voz.



Comandos por gesto:

- Sobe um pouco a página
- Desce um pouco a página
- Trava a página para não utilizar as funções de navegação por gestos
- Destrava a página e volta a utilizar as funções de navegação por gestos
- Direciona para o campo de post e ativa o microfone para criação de novo post por voz
- Após travar a página, desativa a navegação por gestos e a navegação por microfone (se estiver ativa)

Comandos por voz:

- "Suba um pouco" Sobe um pouco a página
- "Desça um pouco" Desce um pouco a página
- "Suba uma página" Sobe o equivalente a uma página no navegador
- "Desça uma página" Desce o equivalente a uma página no navegador
- "Desativar microfone" Desativa o microfone ativo
- "Ir para o topo" Vai para o topo da página
- "Criar novo post" Direciona para o campo de post e ativa o microfone para criação de novo post por voz

Durante a criação de um post:

- "Comando enviar post" Envia o post com o conteúdo escrito
- "Comando desativar microfone" Desativa o microfone ativo
- "Comando limpar texto" Limpa o conteúdo do campo do post

Figuras 20 - Diagramação das adaptações.

Nas próximas seções (Seções [4.2](#) e [4.3](#)) serão apresentadas as adaptações que são realizadas na interface com o usuário e exibidas ao usuário (Adaptação Mãos Livres e Adaptação Comando de Voz) com base nas características/dificuldades dos usuários.

Pesquisa

Você possui LER (Lesão por Esforço Repetitivo)?

Sim
 Não

Você possui alguma limitação ou condição que afete seus membros superiores (por exemplo, braços, mãos)?

Sim
 Não

Você tem alguma condição médica ou deficiência que afeta a mobilidade ou função dos seus membros, como quadriplegia?

Sim
 Não

Você tem alguma condição ou dificuldade que afete a sua capacidade cognitiva?

Sim
 Não

[← Voltar](#) [X Rejeitar](#) [✓ Finalizar](#)

Figura 21 - Modal de pesquisa do perfil do usuário com novas perguntas das novas adaptações.

Tabela 7 - Legenda dos comandos.

Comando	Funcionalidade
<i>up</i>	Sobe um pouco na página.
<i>page-up</i>	Sobe o equivalente a uma página no navegador.
<i>down</i>	Desce um pouco na página.
<i>page-down</i>	Desce o equivalente a uma página no navegador.
<i>top</i>	Vai para o topo da página.
<i>block</i>	Trava a página para não utilizar as funções de navegação por gestos.
<i>release</i>	Destrava a página e volta a utilizar as funções de navegação por gestos.
<i>new-post</i>	Direciona a página para o campo do post e ativa o microfone do campo de post.
<i>dont</i>	Desativa a navegação por gestos e a navegação por microfone (se estiver ativa).
<i>disable-mic</i>	Desativa o microfone ativo.
<i>post-send</i>	Envia o post com o conteúdo escrito.
<i>clear-post-input</i>	Limpa o conteúdo do campo do post.

4.2 Adaptação Mãos Livres

A [Figura 22](#) ilustra as funcionalidades e suas respectivas equivalências visuais dos comandos por gestos.



Figura 22 - Funcionalidades e equivalências visuais por gestos.

Conforme mostra a [Figura 23](#), o módulo de Navegação por Gestos fornece um *feedback* visual por meio do *display* que exibe a câmera do usuário, localizado no canto inferior direito da tela. Esse *feedback* é apresentado em forma de *emoji*, correspondente ao gesto identificado pela biblioteca, a fim de facilitar a visualização dos gestos que estão sendo identificados.

Para auxiliar o usuário a não ter problemas de movimentação, o que pode dificultar a navegação e utilização do sistema, foi implementado um comando de gestos para ‘travar’ a navegação por gestos. Esse comando é identificado pelo gesto de ‘’ seguido do gesto de ‘’, conforme mostrado na [Figura 22](#). Além disso, a funcionalidade de escrever o *post*, que ativa o módulo de Comando de Voz do microfone no campo de postagem, também ‘trava’ a tela, impedindo que o usuário se perca no sistema e/ou tenha problemas. Para liberar a tela, basta fazer o gesto de ‘’ ([Figura 24](#)) que a navegação será ‘destravada/liberada’ e o usuário pode continuar utilizando o sistema normalmente.

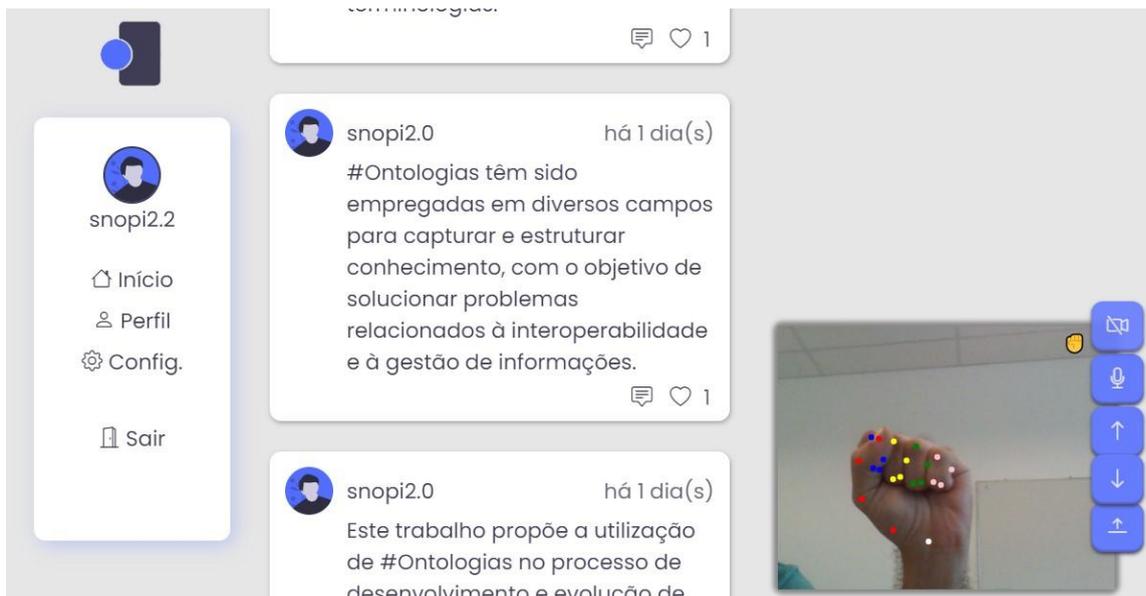


Figura 23 - Print do sistema usando o gesto 'mão fechada'.

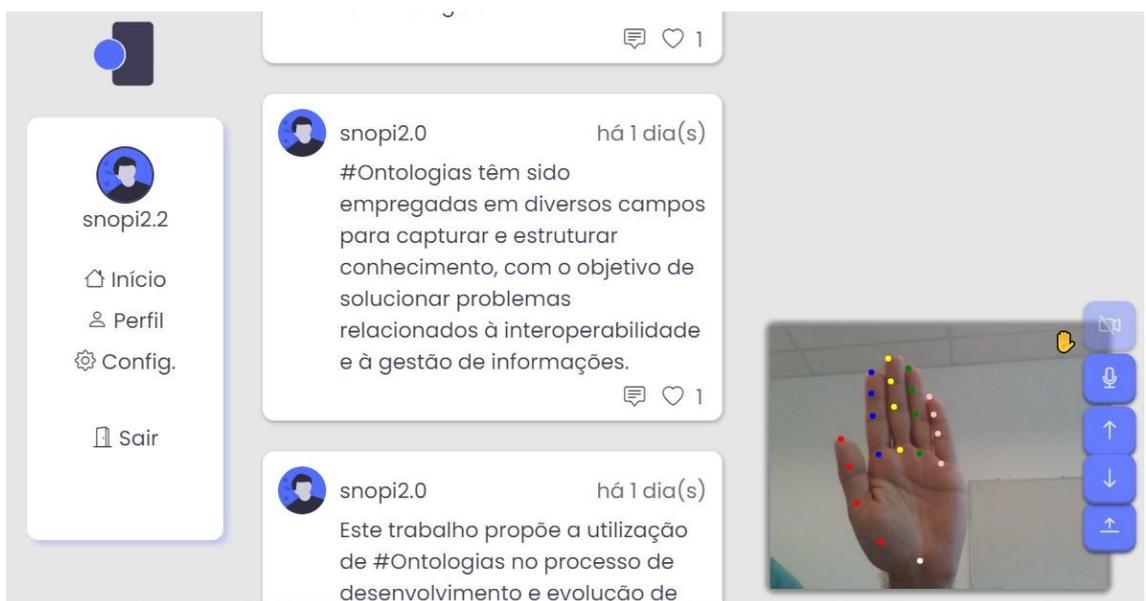


Figura 24 - Print do sistema usando o gesto 'mão aberta'.

4.3 Adaptação Comando de Voz

O módulo de Comandos de Voz possui duas instâncias de microfones diferentes. Quando uma é ativada, apenas ela processa o áudio, e quando a outra é ativada, a instância principal é desativada para que apenas uma funcione por vez. A primeira instância é responsável pelos comandos de navegação, como 'ir para o topo' e 'subir um pouco'. Já a segunda instância é responsável pela atribuição de texto no campo do post e outros comandos, como 'limpar texto' ou 'enviar post'.

O módulo de Comandos de Voz também possui indicadores de uso, assim como o módulo de gestos. Os ícones de cada microfone têm um comportamento de ‘clarear’ e ‘escurecer’ a cor de fundo do botão para mostrar que estão ativos ([Figura 23](#)). Para utilizar o módulo, após estar ativo, só é necessário falar que o sistema identifica e executa o comando. É importante ressaltar que a qualidade da dicção e do áudio do usuário afetam diretamente a capacidade da *API Speech* em reconhecer corretamente o que foi falado. Portanto, quanto mais clara e precisa a dicção e melhor a qualidade do áudio, menor será a probabilidade de a *API Speech* não reconhecer o comando.

Abaixo é apresentada uma imagem ([Figura 25](#)) que ilustra as funcionalidades e suas respectivas equivalências visuais dos comandos de voz.



Figura 25 - Funcionalidades e equivalências visuais por voz.

A [Figura 26](#) ilustra o sistema com a adaptação comando de voz ativada.

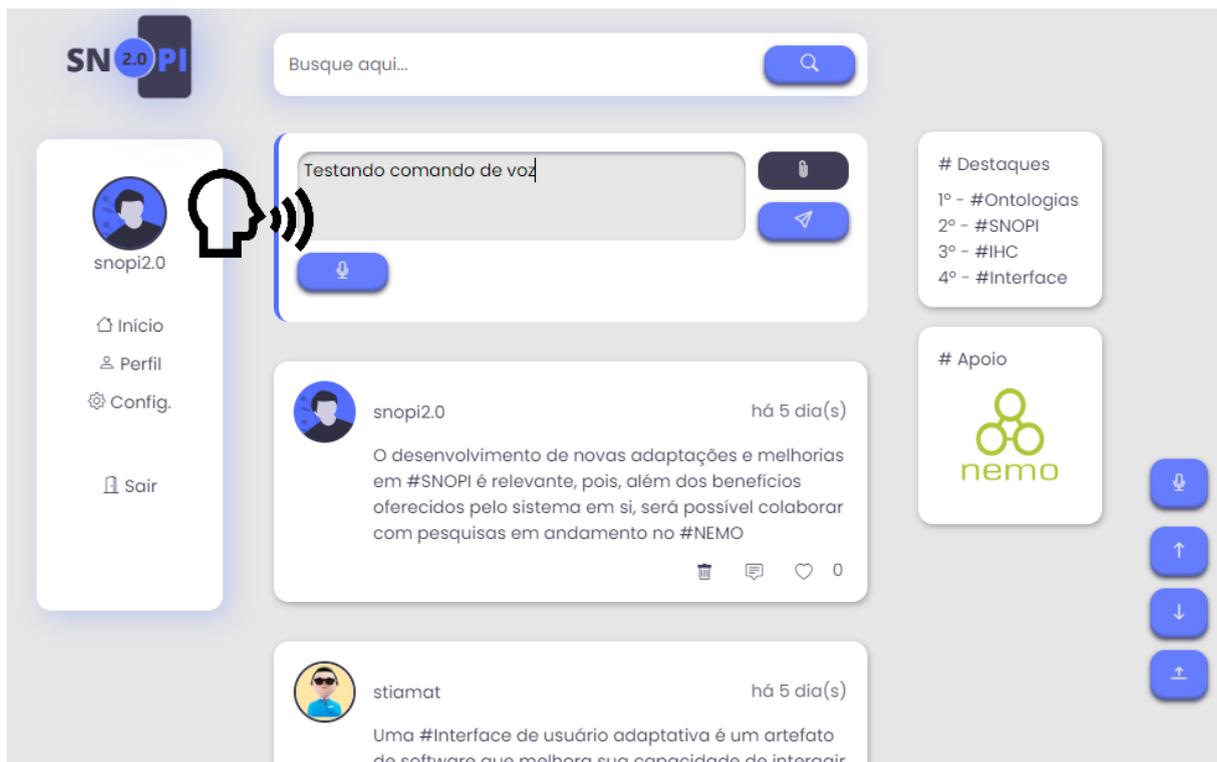


Figura 26 - Print do sistema - adaptação comando de voz.

4.4 Considerações Finais do Capítulo

Este capítulo apresentou as novas adaptações que foram implementadas no sistema SNOPI2.0. Foram apresentadas algumas telas com as novas adaptações, a saber: Adaptação Mãos Livres e Adaptação Comando de Voz. As Novas adaptações implementadas em SNOPI2.0 tiveram o objetivo de melhorar a experiência do usuário e tornar a aplicação mais acessível.

Capítulo 5

Considerações Finais

Neste capítulo são realizadas as considerações finais deste trabalho, sendo apresentadas suas principais contribuições e perspectivas de trabalhos futuros.

5.1 Conclusões

Neste trabalho, foram desenvolvidas novas adaptações no sistema SNOPI2.0, um protótipo de rede social com interface adaptativa. Este trabalho foi realizado no contexto da tese de doutorado do coorientador, que investiga o uso de ontologias para apoiar o desenvolvimento de sistemas com interfaces adaptativas. Este trabalho também está relacionado à tese de doutorado de Costa (2022), na qual foi desenvolvida a HCI-ON (*Human-Computer Interaction Ontology Network*), uma rede de ontologias que aborda aspectos relacionados à IHC e é destinada a apoiar o desenvolvimento de soluções para esse domínio.

A abordagem OADAPT foi utilizada ao longo do desenvolvimento do trabalho para a implementação de novas adaptações no sistema SNOPI2.0. A utilização da abordagem OADAPT permitiu especificar, projetar, implementar e testar as novas adaptações no sistema SNOPI2.0. Como resultado, foram implementadas duas novas adaptações, sendo elas, mãos livres e comando de voz. A primeira adaptação possibilita a navegação no sistema por meio de gestos das mãos e dedos, enquanto a segunda permite aos usuários navegar e fazer postagens no sistema usando apenas comandos de voz. Dessa forma, a abordagem OADAPT se mostrou uma importante contribuição para o desenvolvimento e evolução de sistemas com interface adaptativa.

A utilização de OADAPT para evoluir o SNOPI2.0 permitiu observar necessidades de melhoria na abordagem. Por exemplo, percebeu-se que é preciso deixar mais explícito que uma visão geral das adaptações deve ser estabelecida já na etapa '(i) Identificar o escopo e os usuários do sistema', e que na etapa '(vi) Definir adaptações de interface do usuário', essas adaptações podem ser refinadas considerando-se que a arquitetura do sistema será conhecida, particularmente seu componente de interação com o usuário. Além disso, nessa etapa é que são definidas as regras de adaptações. Assim, seria importante destacar que a escolha das adaptações é uma consequência da identificação das características dos usuários na etapa (i).

Durante a etapa de seleção da ontologia ([etapa iii](#)) e desenvolvimento da ontologia operacional ([etapa vii](#)) a ontologia de referência foi evoluída e utilizada para criar a ontologia operacional ontoSNOPI2.0, a qual foi fundamental para garantir a consistência das novas adaptações implementadas no sistema. A utilização da ontologia operacional também ajudou a definir as regras específicas de como as adaptações seriam implementadas na interface do usuário, garantindo que as adaptações fossem aplicadas de forma consistente.

Durante o desenvolvimento deste trabalho foram aplicados conhecimentos adquiridos ao longo da graduação em diferentes disciplinas. Dentre os principais conhecimentos aplicados, destacam-se aqueles obtidos por meio das seguintes disciplinas:

- **Programação II e III**, que forneceram a base de conhecimento para a lógica de programação, explorando particularidades sobre as linguagens abordadas e apresentando desafios de programação que foram aplicados neste trabalho.
- **Engenharia de Software**, onde foram apresentados conhecimentos sobre aplicação de princípios, métodos e ferramentas para a produção de software de qualidade, considerando aspectos como requisitos do usuário, design, implementação, testes e manutenção.
- **Banco de Dados**, onde foram apresentados conhecimentos básicos sobre criação, manipulação, modelagem e utilização de persistência em bancos relacionais.
- **Projeto de Sistemas de Software**, onde foram apresentados conhecimentos de definir e especificar as melhorias e soluções do projeto de sistema.

No [Capítulo 1](#) desta monografia foram apresentados os objetivos que deveriam ser alcançados por este trabalho. Na [Tabela 8](#) são apresentados os resultados relacionados a cada objetivo ao final do desenvolvimento deste trabalho.

Tabela 8 - Objetivos e seus resultados.

Objetivo	Situação	Resultado
Especificar a evolução nas adaptações a serem realizadas em SNOPI.	Atendido.	Especificação de novas adaptações. (vide Seção 3.1.2).
Projetar a evolução de SNOPI.	Atendido.	Elaboração do projeto de evolução do sistema SNOPI para SNOPI2.0 utilizando a abordagem OADAPT (vide Capítulo 3).
Implementar e testar a evolução projetada.	Atendido.	Adaptações implementadas no sistema SNOPI (renomeado para SNOPI2.0) (vide Seção 3.8 e Capítulo 4).
Fornecer feedback acerca do uso de OADAPT.	Atendido.	Feedback provido ao longo do uso de OADAPT, que foi registrado pelo coorientador deste trabalho e será incorporado à sua pesquisa de doutorado.

5.2 Limitações

Entre as limitações para o desenvolvimento deste trabalho, destaca-se a ausência de documentação adequada de SNOPI, que resultou em dificuldades na compreensão da implementação do código do sistema, o que acarretou em atrasos na implementação das novas adaptações.

O sistema foi testado apenas no navegador Google Chrome. Portanto, embora possa funcionar em outros navegadores, a experiência de uso pode não ser a mesma e podem ocorrer erros devido à falta de suporte a algumas APIs. Por isso, é recomendado utilizar o navegador Google Chrome ou navegadores que utilizam como base o Chromium para garantir uma melhor experiência de uso do sistema. Também, é importante destacar, que as novas adaptações não funcionam na versão mobile.

No início do projeto, foi idealizada a adaptação *Dynamic Mode*, que consistia em ativar o modo escuro de acordo com a localização/horário do equipamento do usuário, porém, ela não pôde ser implementada devido a questões técnicas no código do sistema. Em vez disso, foi adicionada ao sistema a opção de configuração do perfil do usuário para ativar, desativar ou deixar automático o modo escuro.

Também não foi possível desenvolver testes unitários por questões técnicas. O projeto contém dois gerenciadores de pacotes (Maven e NPM/YARN) e, a partir de

cada um deles foram adicionados muitos pacotes quando o projeto foi implementado em sua primeira versão. Dentre esses pacotes, muitos estão desatualizados e contêm mudanças que podem ‘quebrar’ o sistema. Os testes unitários precisaram ser removidos, pois a biblioteca de testes não estava funcionando como deveria. À medida que alguns pacotes eram necessários, eles iam sendo atualizados para que as novas adaptações pudessem ser implementadas e funcionarem corretamente.

5.3 Trabalhos Futuros

Como trabalhos futuros, os seguintes pontos foram identificados como possíveis melhorias no sistema:

- Adicionar novos gestos e/ou comandos de voz com funcionalidades para acessar o menu de configurações.
- Adicionar novos gestos e/ou comandos de voz para manipular os posts do *feed* do usuário (excluir, comentar, curtir).
- Adicionar uma adaptação que utilize os dados de utilização do usuário, como verificar que ele interage mais fácil de um lado da tela, então ajustar o *layout* para deixar mais simples para o usuário.
- Melhorar a confiabilidade do Classificador do *FingerPose*, para captar gestos com maior acurácia.
- Coletar o *feedback* dos usuários sobre as novas adaptações.
- Monitorar a utilização do usuário da adaptação Mãos Livres e deixar travada a tela para movimentos indesejados enquanto ele não demonstra vontade de interagir com o site ativamente (*e.g.*, o usuário lendo um *post* e passa a mão na cabeça fazendo sinal de ‘👍’ inconscientemente).
- Aumentar o número de adaptações possíveis.
- Melhorar a documentação do projeto base SNOPI para facilitar a manutenção.
- Implementar uma funcionalidade que rastreie a localização dos cliques do usuário durante a utilização do sistema. E, com base nesses dados, definir quais adaptações podem ser feitas para melhorar a experiência do usuário, como por exemplo, verificar quantas vezes ele voltou a uma

página, identificar se o usuário teve dificuldade ao utilizar o sistema e analisar seu comportamento no sistema.

Referências Bibliográficas

- ABOU-ZAHRA, S.; SINCLAIR, N. User Stories from the Web. W3C Web Accessibility Initiative (WAI), 2017. Disponível em: <https://www.w3.org/WAI/people-use-web/user-stories/#reporter>. Acesso em: 14 jun. 2023.
- ARAÚJO, Ricardo José de. Interfaces Adaptativas para Daltônicos com base em Ontologias. Dissertação de Mestrado. Faculdade Campo Limpo Paulista (FACCAMP), Campo Limpo Paulista, 2017. Disponível em: <https://www.cc.faccamp.br/Dissertacoes/RicardoJoseAraujo.pdf>. Acesso em: 15 dez. 2022.
- BENYON, David; MURRAY, Dianne. Experience with adaptive interfaces. *The Computer Journal*, v. 31, n. 5, p. 465-473, 1988.
- CHEN, Liutao; DING, Gangyi; JIN, Qiankun. Research on Virtual Range multi-domain modeling methodology based on Meta-model. In: 2010 2nd International Conference on Computer Engineering and Technology. IEEE, 2010. p. V4-176-V4-180.
- COOPER, Alan. The inmates are running the asylum. In: *Software-Ergonomie'99*. Vieweg+ Teubner Verlag, Wiesbaden, 1999. p. 17-17.
- COSTA, Simone Dornelas. An Ontology Network to support Knowledge Representation and Semantic Interoperability in the HCI Domain. Tese de Doutorado. Programa de Pós-Graduação em Informática (PPGI), Universidade Federal do Espírito Santo (UFES), Vitória-ES, Brasil, 2022. Disponível em: https://nemo.inf.ufes.br/wp-content/papercite-data/pdf/an_ontology_network_to_support_knowledge_representation_and_semantic_interoperability_in_the_hci_domain_2022.pdf. Acesso em: 20 dez 2022.
- COSTA, S. D., BARCELLOS, M. P., FALBO, R. D. A., & CASTRO, M. V. H. B. (2020, November). Towards an ontology network on human-computer interaction. In *International Conference on Conceptual Modeling* (pp. 331-341). Springer, Cham.
- FALBO, Ricardo A.: Engenharia de Requisitos: Notas de aula. 2017. Departamento de Informática, Universidade Federal do Espírito Santo, Vitória-ES. Disponível em:

http://www.inf.ufes.br/~jssalomon/wp-content/uploads/disciplinas/engreq/Notas_Aula_Engenharia_Requisitos.pdf. Acesso em: 26 jun 2023.

- FALBO, R. de A. *et al.* Ontology patterns: clarifying concepts and terminology. In: Proceedings of the 4th Workshop on Ontology and Semantic Web Patterns. 2013.
- FREITAS, Alexandre Adler C.; SCALSER, Murilo B.; COSTA, Simone Dornelas; BARCELLOS, Monalessa P. (2022). Towards an Ontology-based Approach to Develop Software Systems with Adaptive User Interface. In: Proceedings of the 21st Brazilian Symposium on Human Factors in Computing Systems (pp. 1-7). <https://dl.acm.org/doi/abs/10.1145/3554364.3559139>
- GRAY, Dave; BROWN, Sunni; MACANUFO, James. Gamestorming: A playbook for innovators, rulebreakers, and changemakers. " O'Reilly Media, Inc.", 2010.
- GUIZZARDI, Giancarlo. On Ontology, ontologies, Conceptualizations, Modeling Languages, and (Meta) Models. In: Proceedings of the 2007 conference on Databases and Information Systems IV: Selected Papers from the Seventh International Baltic Conference DB&IS'2006. 2007. p. 18-39.
- HAPPEL, Hans-Jörg; SEEDORF, Stefan. Applications of ontologies in software engineering. In: Proc. of Workshop on Semantic Web Enabled Software Engineering" (SWESE) on the ISWC. 2006. p. 5-9.
- ISO 9241-210, 2010. Ergonomics of human-system interaction - Part 210: Human-centred design for interactive systems. International Organization for Standardization.
- JAMESON, Anthony. Adaptive interfaces and agents. In: The human-computer interaction handbook. CRC press, 2007. p. 459-484.
- KULTSOVA, Marina; POTSELUICO, Anastasiya; DVORYANKIN, Alexander. Ontology Based Personalization of Mobile Interfaces for People with Special Needs. In: Conference on Creativity in Intelligent Technologies and Data Science. Springer, Cham, 2019. p. 422-433. https://doi.org/10.1007/978-3-030-29750-3_33
- LANGLEY, Pat. User modeling in adaptive interface. In: UM99 User Modeling. Springer, Vienna, 1999. p. 357-370.
- LIU, Chi-Lun; HUANG, Hsieh-Hong. Ontology-Based Adaptive and Customizable Navigation Method in Online Retailing Websites. In: International Conference

- on HCI in Business, Government, and Organizations. Springer, Cham, 2016. p. 228-237.
- NIELSEN, Jakob. Usability engineering. Morgan Kaufmann, 1994.
- OPPERMANN, Reinhard (Ed.). Adaptive user support: ergonomic design of manually and automatically adaptable software. CRC Press, 1994.
- PRESSMAN, R. S.; LOWE, D. Web engineering: a practitioner's approach. [S.l.]: McGraw-Hill Education, 2009. v. 1.
- RUY, Fabiano Borges; FALBO, Ricardo de Almeida; BARCELLOS, Monalessa Perini; COSTA, Simone Dornelas; GUIZZARDI, Giancarlo. SEON: A software engineering ontology network. In: Knowledge Engineering and Knowledge Management: 20th International Conference, EKAW 2016, Bologna, Italy, November 19-23, 2016, Proceedings 20. Springer International Publishing, 2016. p. 527-542.
- SCALSER, M. B. SNOPI: Um Sistema de Interface Adaptativa baseada em Ontologia. TCC (Graduação) - Curso Ciência da Computação, Universidade Federal do Espírito Santo (UFES), Vitória, 2022. Disponível em: <https://dev.nemo.inf.ufes.br/hcion/paper-pdf/scalser2022.pdf>. Acesso em: 10 dez. 2022.
- SCAPIN, Dominique L. Organizing human factors knowledge for the evaluation and design of interfaces. International Journal of Human-Computer Interaction, v. 2, n. 3, p. 203-229, 1990.
- SCHERP, Ansgar; SAATHOFF, Carsten; FRANZ, Thomas; STAAB, Steffen. Designing core ontologies. Applied Ontology, v. 6, n. 3, p. 177-221, 2011.
- SCHNEIDER-HUFSCHMIDT, Matthias; MALINOWSKI, Uwe; KUHME, Thomas (Ed.). Adaptive user interfaces: Principles and practice. Elsevier Science Inc., 1993.
- SOUI, Makram *et al.* An ontology-based approach for user interface adaptation. In: Advances in Intelligent Systems and Computing. Springer, Cham, 2017. p. 199-215. https://doi.org/10.1007/978-3-319-45991-2_13
- SUÁREZ-FIGUEROA, Mari Carmen *et al.* Introduction: Ontology engineering in a networked world. In: Ontology engineering in a networked world. Springer, Berlin, Heidelberg, 2012. p. 1-6.