

AMPAREE – Ambiente Multidisciplinar de Pesquisa Auxiliando os Resultados e o Ensino de Enfermagem: Criação do constructo e abordagens iniciais

Maria Eduarda Eira da Silva^{1*}, Maria Fernanda Lima Cena², Samara de Melo Araújo³, Letícia Do Nascimento Silva⁴, Maria Luiza Da Silva Viana⁵, Rebeca Santos Silva⁶, Jadson Freire da Silva⁷

¹Graduada em Análise e Desenvolvimento de Sistemas, Centro Universitário Brasileiro, Brasil.

²Graduanda em Enfermagem, Centro Universitário Brasileiro, Brasil.

³Graduanda em Administração, Centro Universitário Brasileiro, Brasil.

⁴Graduanda em Enfermagem, Centro Universitário Brasileiro, Brasil.

⁵Graduanda em Enfermagem, Centro Universitário Brasileiro, Brasil.

⁶Graduanda em Enfermagem, Centro Universitário Brasileiro, Brasil.

⁷Bacharel em Administração, Doutor em Desenvolvimento e Meio Ambiente, Universidade Federal de Pernambuco, Brasil. (*Autor correspondente: jadsonfreireufpe@hotmail.com)

Histórico do Artigo: Submetido em: 23/11/2024 – Revisado em: 22/03/2025 – Aceito em: 10/05/2025

RESUMO

A importância de se estudar e aprender sobre os conceitos básicos de enfermagem se apresenta como um grande ponto de estímulo para os estudantes da saúde em âmbitos técnicos ou de graduação. A gama de informações que podem ser obtidas por intermédio de sítios diversos fazem a necessidade de organizar esses dados em um ambiente virtual de aprendizagem. Frente a isso, o objetivo desse projeto se versa na criação de um ambiente virtual de aprendizado multidisciplinar em linguagem python para enfermagem enquanto abordagens iniciais e criação do constructo do tema. A partir disso visa-se apresentar toda a composição inicial do ambiente virtual enquanto componente teórico e estrutural e as primeiras aproximações de um produto tangível que possibilite os primeiros testes frente a teoria em saúde, palavras chaves relevantes e o constructo do tema consolidado para outras proposições.

Palavras-chave: python, dashboard, criação de constructo, saúde, enfermagem

AMPAREE - Multidisciplinary Research Environment Supporting Nursing Outcomes and Education: Creation of the construct and initial approaches

ABSTRACT

The importance of studying and learning about the basics of nursing is a great stimulus for health students in technical or undergraduate settings. The range of information that can be obtained from various sites makes it necessary to organize this data in a virtual learning environment. In view of this, the aim of this project is to create a multidisciplinary virtual learning environment in Python language for nursing as an initial approach and creation of the subject construct. The aim is to present the entire initial composition of the virtual environment as a theoretical and structural component and the first approximations of a tangible product that enables the first tests against health theory, relevant keywords and the consolidated theme construction for other propositions.

Keywords: python, dashboard, construct creation, health, nursing.

1. Introdução

Desafios observados na pandemia foram alvos de grandes discussões e manejo das esferas sociais, governamentais e individuais. Verificou-se que o novo corona vírus (COVID-19) trouxe consigo problemas que não foram somente voltados a doença, mas também estruturais em diferentes frentes e que esses problemas, latentes, potencializaram a doença no território junto as condições socioambientais e ecossistêmicas¹². Nesse tempo, os profissionais e estudantes de saúde tiveram que redobrar sua atenção no aprendizado das patologias



e como elas se manifestam no meio ambiente, de modo a aumentar a assertividade dos diagnósticos e amparar em potencial os enfermos.

Assim, aumenta a necessidade de ambientes que consigam entregar para o profissional da saúde subsídios para o seu aprimoramento enquanto conhecimento. O estudante técnico, da graduação e o egresso que exerce a profissão nos diferentes ambientes do Brasil mantém e deve manter um aprendizado contínuo e assim a pesquisa como estratégias que facilitem o aprendizado em diferentes lugares se incluem como grandes aliadas na educação em saúde³⁴. A linguagem de programação se inclui a isso, dando propriedade para criação de softwares responsivos que entregam para os operadores mecanismos eficientes e eficazes para obtenção de conteúdo de interesse⁵.

Frente a isso, é relevante não somente a criação de ambientes virtuais, mas ambientes que auxiliem no aprendizado de uma gama de estudantes que serão peças centrais na promoção da qualidade de vida de e saúde da população. Desta forma, observa-se conceitos multidisciplinares dialogando para não somente verificar o organizacional dos conceitos em saúde, mas também como criar um local adequado para o respectivo aprendizado.

De acordo com a Instituto de Estudos em Saúde Coletiva⁶, define-se a Saúde coletiva como um grande campo que se volta a discutir e promover as relações da sociedade frente a saúde, relacionando motivações que ultrapassam a esfera da saúde, encontrando os conceitos socioeconômicos, ambientais, demográficos, epidemiológicos e culturais.

Os escritos supracitados reverberam e concomitam com a proposição de um ambiente virtual em linguagem de programação em python em prol para o ensino multidisciplinar da saúde e da enfermagem. Para desenvolvimento de estudos que trazem consigo conteúdos de saúde, é necessário um amparo de informações oriundas de livros clássicos sobre diagnósticos. Estes desafios justificam a necessidade da relação da Saúde Coletiva com outras ciências, de modo a explicar com maior acurácia problemáticas pertinentes a regiões de interesse.

Sendo assim, a pesquisa tem como objetivo apresentar os primeiros arcabouços teóricos e constructos de composição da criação de um ambiente virtual de aprendizagem em linguagem de programação em python para o ensino e capacitação em enfermagem, denominado de AMPAREE (Ambiente Multidisciplinar de Pesquisa Auxiliando os Resultados e o Ensino de Enfermagem).

Os objetivos específicos abarcam a apresentação dos conceitos sobre a saúde e a linguagem de programação aplicadas e correlacionadas; a observação dos layouts que entreguem aos estudantes subsídios informativos através da responsividade; as proposições de reflexões acerca da educação e conscientização na saúde como agente mitigador e potencializador de novas contribuições e o amparo sobre as considerações territoriais sobre o planejamento, política e gestão em Saúde.

2. Material e Métodos

Uma vez tratando-se de um arcabouço teórico e criação de constructo, não existe uma área de estudo física definida. No entanto, é necessário um amparo profundo da teoria e diagnósticos em saúde e sobretudo em enfermagem, de modo a apresentar os principais pontos que podem ser obtidos e seccionados para um ambiente virtual de aprendizagem eficaz para contribuir na educação e na capacitação de estudantes e profissionais.

Apresentações de constructos e informações basilares sobre um determinado tema não é algo novo na ciência, sendo revelado por outras pesquisas que tiveram o anseio de apresentar documentações iniciais sobre um determinado tema em questão⁷. Vale ressaltar que mesmo sendo imposto em um tema ou uma área que é estabelecida, a inovação é instituída sobre a remodelação de layouts, produtos ou serviços novos ou remodelados de modo a entregar novas capacidades e usabilidades sobre os seus consumidores – usuários.

Portanto, mesmo havendo na pesquisa informações documentais ou documentadas, ela é a iniciação de um processo inovador em programação sobre a enfermagem e seus processos educacionais em questão^{8,9}.

Sendo assim, além dos constructos abordados como informações basilares para a criação do programa em si, uma plataforma de processamento de informações em linguagem python foi escolhida sendo o PyCharm (versão colaborativa), programa gratuito que auxilia na escrita e testes em diversas linguagens de programação, sobretudo a Python. Além disso, a literatura em Ciências da Saúde escolhida para apresentação do constructo e proposições foi a NANDA (North American Nursing Diagnosis Association)¹⁰ sendo uma publicação oficial que reúne os diagnósticos de enfermagem reconhecidos internacionalmente, organizados de forma padronizada.

3. Resultados e Discussão

3.1 Proposição dos constructos iniciais

A utilização da informática, programação, ciência da informação ou administração dos sistemas de informação na saúde não é novo, sendo observado suas implementações ao longo da história. Frente a essas considerações, apresenta-se o constructo e suas composições acerca do AMPAREE:

Proposição 1: O AMPAREE (Ambiente Multidisciplinar de Pesquisa Auxiliando os Resultados e o Ensino de Enfermagem) é um conceito que deriva em um software (programa) [ou softwares] que tem como proposta auxiliar o acesso à informação pré-estabelecida na pesquisa e ciência da saúde - enfermagem. O programa não diagnostica nem substitui um profissional de saúde capacitado para observar as premissas de entrada e de saída mediante processamento;

Proposição 2: O NANDA¹⁰, importante base de dados em saúde será o arcabouço central para obtenção das informações relevantes de estímulo resposta; isto é, as entradas e as saídas terão como base o livro em questão e a necessidade futura de possíveis atualizações para inclusão e exclusão de diagnósticos;

Proposição 3: Mesmo havendo como literatura basilar o NANDA, recomendações governamentais e específicas de cada estado brasileiro poderão ser integradas ao código, como forma de respeitar as recomendações solicitadas ao território em questão;

Proposição 4: Sabendo da existência de outras bases de dados em saúde de referência, levanta-se a necessidade de uma integração entre conceitos e conteúdo ou a criação de outra plataforma de apoio a saúde – ciências da saúde – no que se refere a enfermagem e seus correlatos;

Proposição 5: Inicialmente, pensou-se na criação de uma calculadora com duas colunas de entrada centrais. Esse pensamento se deu devido ao ambiente de consulta que os pontos centrais são ouvidos e desta forma, incluídos em ambiente plataforma.

Sendo assim, recomenda-se um programa que mantenha um ou dois ambientes de entrada para uma melhor performance do software; a redução de colunas de palavras chaves levanta a necessidade de códigos mais robustos de obtenção de informação por semelhança, já o aumento de colunas pode reduzir essa necessidade por seccionamento de informações de interesse. Em ambos os casos, é possível elevar a qualidade do acesso à informação por parte do programa mediante recebimento dos dados iniciais;

Proposição 6: A busca pela verossimilhança das informações próximas é contínua. Modelos em máquina podem melhorar na compreensão do que será escrito na entrada do AMPAREE e um bom resultado disso

reverbera no melhor entendimento dos conceitos por parte do discente. Sendo assim, seja para o programa em duas colunas de entrada inicial ou entrada única, é necessário que os entendimentos não sejam literais para um resultado assertivo.

Proposição 7: A responsividade e a capacidade de exposição, de modo leve, em plataformas desktop ou mobile é necessária. Tendo em vista que é um software educacional, estudantes utilizarão o supracitado para melhores compreensões diagnósticas, comparação entre fatores e resultados a serem feitos após diagnósticos implementados. Devido a esses motivos, não se sabe exatamente o grau de estrutura socioeconômica por parte dos discentes, de modo a entregar um ambiente que consiga ser possível a execução em qualquer dispositivo, seja ele o mais simples ou o mais arrojado.

Proposição 8: O AMPAREE é modular no que se refere a sua produção e construção. A quantidade elevada de diagnósticos por parte do NANDA e a divisão do mesmo em domínios facilita esse constructo. Desta forma é possível criar subprogramas derivados de um todo (a exemplo, AMPAREE-ATIVIDADE/REPOUSO [domínio 4]), dando condições de uma gama de pesquisadores participarem do projeto. Há também a possibilidade de criar um AMPAREE-NOC/NIC com informações específicas que a enfermagem utiliza com propriedade e com frequência.

Proposição 9: Recomendar-se-á uma interface que consiga satisfazer os interesses dos discentes, principalmente. Essa interface deverá ser atualizada mediante as necessidades dos próprios e os docentes deverão atualizar-se frente a essas demandas. O sentido educativo de um aprendizado eficaz sobressairá o engessamento do programa mediante o tempo de implementação; neste fato, a atualização visual deverá acompanhar as mudanças semióticas de modo a atrair novos (e antigos) profissionais e estudantes das ciências da saúde.

Proposição 10: Independentemente da linguagem escolhida, a proposta da manutenção do estímulo resposta a partir dos possíveis sintomas com base em um documento com confiabilidade existe e emerge para o auxílio no ensino das ciências da saúde, em especificidade, na enfermagem, sejam em níveis técnicos ou superior. A escolha do Python na produção inicial do software deu-se pela familiaridade dos operadores a língua; no entanto, caso haja futuramente linguagens com respostas eficientes, recomenda-se a aplicação e implementação a posteriori.

Assim, apresenta-se o constructo através de um código em Python que utiliza a biblioteca Tkinter para criar uma interface gráfica de usuário de uma calculadora de combinações. A calculadora faz correspondências de entrada com um conjunto de dados pré-definido e, com base nessas correspondências, calcula resultados e recomendações.

O código começa importando as bibliotecas necessárias, define um dicionário de dados e um dicionário de recomendações. Posteriormente, apresenta-se há duas funções definidas: uma para encontrar correspondências próximas e outra para calcular os resultados e as recomendações com base na entrada do usuário.

Depois disso, a interface gráfica é configurada com um campo de entrada, um botão de cálculo e dois rótulos para exibir os resultados e as recomendações. Observa-se que o AMPAREE é formado por proposições e constructos simplificados para que haja uma execução limpa e um produto próximo aos resultados esperados nos diagnósticos publicizados em ambiente científico.

A biblioteca Tkinter é a interface padrão do Python para a criação de interfaces gráficas (GUI). Ela fornece um conjunto de ferramentas e widgets que permitem aos desenvolvedores criarem aplicativos com janelas, botões, campos de texto, menus, entre outros elementos visuais, de forma simples e eficiente. Tkinter

é uma camada de interface para a biblioteca Tcl/Tk, que é uma linguagem de script embutida e um sistema de janelas multiplataforma.

Já a biblioteca difflib é uma biblioteca padrão do Python utilizada para comparar sequências, especialmente úteis para comparar e produzir diferenças entre strings e listas de strings. Ela pode ser usada para encontrar similaridades, criar diffs (diferenças) entre textos e gerar relatórios de comparação de forma detalhada.

3.2 Proposição dos códigos basilares

Frente aos constructos, abaixo, apresenta-se os códigos basilares do processo (Código 1)

Código 1 – Informações basilares do AMPAREE-Dianósticos

```
import tkinter as tk
from tkinter import ttk
import difflib

# Dicionário de dados
data = {
    "ambiente sem condições sanitárias": "Manutenção do lar prejudicada",
    "capacidade prejudicada de manter a moradia": "Manutenção do lar
prejudicada",
    "dificuldade para manter um ambiente confortável": "Manutenção do lar
prejudicada",
    "equipamento de cozinha insuficiente": "Manutenção do lar prejudicada",
    "equipamento insuficiente para a manutenção da casa": "Manutenção do lar
prejudicada",
    "excesso de responsabilidades familiares": "Manutenção do lar prejudicada",
    "falta de roupas de cama": "Manutenção do lar prejudicada",
    "padrão de doenças causadas por condições não higiênicas": "Manutenção do
lar prejudicada",
    "padrão de infecções causadas por condições não higiênicas": "Manutenção do
lar prejudicada",
    "roupas insuficientes": "Manutenção do lar prejudicada",
    "solicitação de ajuda para a manutenção da casa": "Manutenção do lar
prejudicada",
    "Falta de adesão a atividades de saúde": "Autonegligência",
    "Higiene ambiental insuficiente": "Autonegligência",
    "Higiene pessoal insuficiente": "Autonegligência",
}

# Dicionário de recomendações
recommendations = {
    "Manutenção do lar prejudicada": "Auxílio na limpeza",
    "Autonegligência": "Mais foco",
}

# Função para encontrar todas as correspondências acima de um limiar
def find_closest_matches(input_str, choices, cutoff=0.4):
    # Procura correspondências de substring em todas as escolhas
    substring_matches = [choice for choice in choices if input_str.lower() in
choice.lower()]
```

```

# Retorna as correspondências de substring que atendem ao limiar de corte
return [match for match in substring_matches if
        difflib.SequenceMatcher(None, input_str.lower(),
match.lower()).ratio() >= cutoff]

# Função para calcular os resultados e as recomendações
def calculate():
    # Obtém a entrada do usuário e remove espaços em branco desnecessários
    user_input = entry.get().strip()

    # Verifica se a entrada do usuário está vazia
    if not user_input:
        result_label.config(text="Por favor, insira uma entrada válida")
        recommendation_label.config(text="")
        return

    # Divide a entrada do usuário por vírgulas e remove espaços desnecessários
    keywords = [keyword.strip() for keyword in user_input.split(',')]

    # Inicializa listas para armazenar resultados e recomendações
    possible_results = set()
    recommendations_list = set()

    # Encontra todas as correspondências próximas e como substrings para cada
    palavra-chave
    for keyword in keywords:
        col_matches = find_closest_matches(keyword, data.keys())

        # Obtém os resultados e recomendações para cada correspondência
    encontrada
        for match in col_matches:
            result = data.get(match)
            if result:
                possible_results.add(result)
                recommendation = recommendations.get(result, "Recomendação não
    encontrada")
                recommendations_list.add(recommendation)

    # Verifica se foram encontrados resultados
    if not possible_results:
        result_label.config(text="Nenhum resultado encontrado")
        recommendation_label.config(text="")
        return

    # Exibe os resultados e recomendações na interface gráfica
    results_text = "Possíveis diagnósticos:\n" + "\n".join(possible_results)
    recommendations_text = "Recomendações:\n" + "\n".join(recommendations_list)
    result_label.config(text=results_text)
    recommendation_label.config(text=recommendations_text)

# Configuração da interface gráfica
root = tk.Tk()
root.title("Calculadora de Combinações")

```

```
# Aba de entrada
label = ttk.Label(root, text="Entrada (separar palavras-chave por vírgula)")
label.pack(pady=10)
entry = ttk.Entry(root)
entry.pack(pady=10)

# Botão de calcular
calculate_button = ttk.Button(root, text="Gerar Diagnóstico", command=calculate)
calculate_button.pack(pady=10)

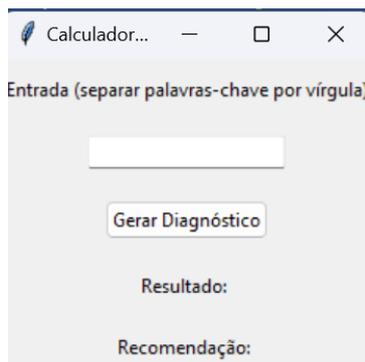
# Label de resultado
result_label = ttk.Label(root, text="Resultado: ")
result_label.pack(pady=10)

# Label de recomendação
recommendation_label = ttk.Label(root, text="Recomendação: ")
recommendation_label.pack(pady=10)

# Inicia o loop da interface gráfica
root.mainloop()
```

Nota-se que o pensamento proposto como base condiciona as respostas mediante as e os condicionantes expostos para procura. A exposição mantém característica específica e intuitiva (Figura 1)

Figura 1 – Imagem correspondente ao front dos códigos



Entende-se que principalmente o *frontend* deve ser ajustado. No entanto, os resultados que podem ser observados quando forem alimentados por completo o banco de dados de enfermagem poderá contribuir ativamente o ensino como um todo. A Figura 2 e 3 apresentam uma aplicação prática do programa em execução.

Figura 2 – Aplicação prática do AMPAREE com uma variável

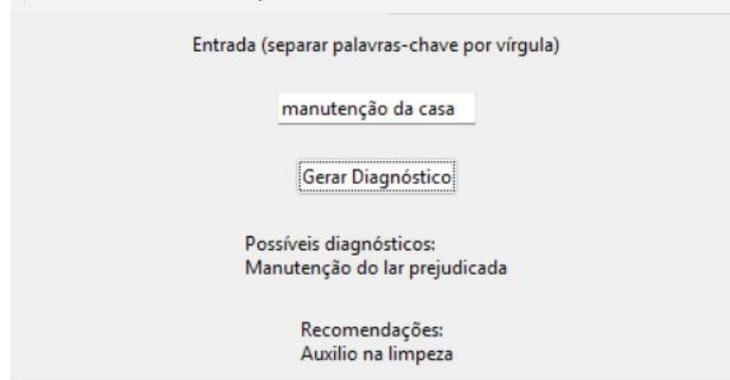
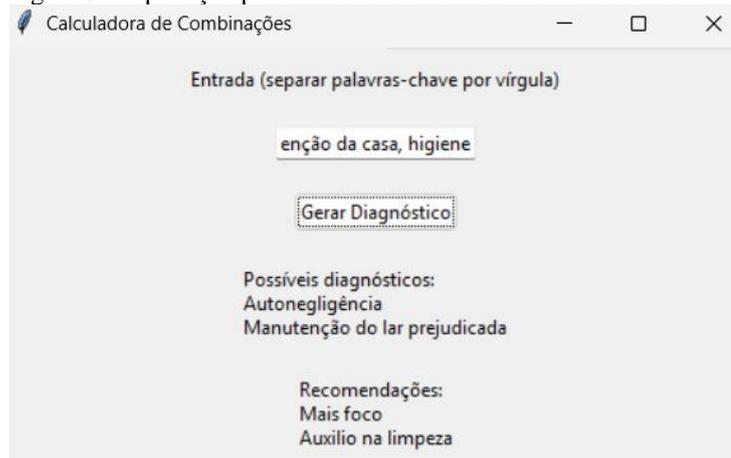


Figura 3 – Aplicação prática do AMPAREE com duas variáveis



A Figura 2 demonstra a aplicação de uma variável inserida no AMPAREE; sob perspectiva que o estudante esteja em um estudo de caso em enfermagem e depara-se com problemáticas que envolvem variáveis possíveis de resumir em palavras chaves, o software retornará com os possíveis diagnósticos e as possíveis recomendações, tendo como base o NANDA mais atual. Isso dará para o estudante rápida capacidade de checar veracidade ou variedade de diagnósticos sobre uma determinada patologia.

A Figura 3 apresenta a variação de inserção de palavras chaves, dando maior complexidade ao programa em retornar informações relevantes. Enquanto na Figura 2 aplicou-se apenas uma palavra (manutenção da casa) a 3 apresentou duas condições (manutenção da casa e higiene), ampliando a margem de um diagnóstico, esse podendo ser mais consistente mediante a necessidade do paciente ou exercício-problema.

A alimentação de um banco de dados consistente é um desafio a se combater uma vez que a quantidade de variáveis, informações, diagnósticos, recomendações e outros por menores entregam uma gama de dados que devem ser tabulados e dispostos de modo a integrar com outros ambientes caso seja necessário.

4. Conclusão

A criação de um constructo implementa as condições necessárias para o desenvolvimento de um programa que auxilie o sistema educacional no que se refere a enfermagem. A entrega de um sistema de códigos que podem abrir precedentes de possibilidade para a construção de um front – exposição de um local

que por palavras chaves um possível diagnóstico poderá auxiliar na educação e no aprendizado do enfermeiro e enfermeira.

É necessário à validação dos códigos e apresentação dos mesmos a comunidade de enfermagem, de modo a melhorar ainda mais e aproximar as demandas diárias dos alunos junto aos seus anseios. Levanta-se a necessidade de novas modulações, estas sendo respostas das melhorias aplicadas e implementadas no próprio software, desenvolvendo ambientes de fáceis manipulações e usos contínuos.

É necessário novas aplicações dos códigos em ambientes de outras linguagens. Essa possibilidade poderá entregar a melhor performance no que tange a responsividade do conteúdo, melhor usabilidade e disseminação. A proposição do constructo e suas primeiras impressões foram estabelecidas, no entanto, as limitações da pesquisa são presentes sendo importante a continuidade da mesma pela comunidade científica.

5. Referências

1. Freire-Silva J, et al. A utilização do planejamento territorial no combate da COVID-19: considerações sobre a situação dos leitos nos municípios de Pernambuco, Brasil. *Vigil Sanit Debate Soc Ciênc Tecnol.* 2020;8(2):16-27.
2. Sobral MFF, et al. A retrospective cohort study of 238,000 COVID-19 hospitalizations and deaths in Brazil. *Sci Rep.* 2022;12(1):3629.
3. Oguisso T, Campos PFS. Por que e para que estudar história da enfermagem?. *Enferm Foco.* 2013;4(1):49-53.
4. Carrijo CIS, Pontes DO, Barbosa MA. Reflexão sobre a importância da temática saúde da família no ensino da graduação em enfermagem. *Rev Bras Enferm.* 2003;56:155-9.
5. Malucelli A, et al. Sistema de informação para apoio à Sistematização da Assistência de Enfermagem. *Rev Bras Enferm.* 2010;63:629-36.
6. Instituto de Estudos em Saúde Coletiva. Definição de Saúde Coletiva. Disponível em: <https://www.iesc.ufrj.br/>
7. Silva, Jadson Freire, Marco Aurélio Benevides de Pinho, and Brigitte Renata Bezerra de Oliveira. "Coworkings Sociais como apoio ao empreendedorismo: proposição de definição do construto e de abordagem de investigação." *Navus: Revista de Gestão e Tecnologia* 12 (2022): 29.
8. Ziemnowicz, Christopher. "Joseph A. Schumpeter and innovation." *Encyclopedia of creativity, invention, innovation and entrepreneurship.* Cham: Springer International Publishing, 2020. 1517-1522.
9. Gilbert, Richard. "Looking for Mr. Schumpeter: where are we in the competition--innovation debate?." *Innovation policy and the economy* 6 (2006): 159-215.
10. Herdman TH, et al. Diagnósticos de enfermagem da NANDA-I: definições e classificação 2021-2023. Porto Alegre: Artmed; 2021. Acesso em: 23 jan. 2025.