



# Concepção e implementação de um *wizard* para análise de grafos de conhecimentos temporais

*Caio Henrique Pardal Dos Santos*      *Julio C. dos Reis*

Relatório Técnico - IC-PFG-22-22  
Projeto Final de Graduação  
2022 - Julho

UNIVERSIDADE ESTADUAL DE CAMPINAS  
INSTITUTO DE COMPUTAÇÃO

The contents of this report are the sole responsibility of the authors.  
O conteúdo deste relatório é de única responsabilidade dos autores.

# Concepção e implementação de um *wizard* para análise de grafos de conhecimentos temporais

Caio Henrique Pardal Dos Santos, Julio Cesar dos Reis \*

## Resumo

*TKGEvolViewer* [1] é uma ferramenta para a visualização interativa da evolução de grafos de conhecimento. Esse software possibilita a exploração gráfica de *Temporal Knowledge Graphs* (TKGs) [2] a partir de valores de métricas codificadas nas estruturas dos TKGs. A ferramenta de software permite uma forma de explorar esses grafos para seu entendimento e compreensão de métricas sobre os conceitos codificados. Com o uso da ferramenta, pessoas podem conduzir análises visuais de *TKGs* através da escolha de uma métrica. Neste trabalho, refinamos a ferramenta por meio da concepção e implementação de um módulo de um Wizard interativo, que apoia usuários na condução guiada de análises. A solução construída permite uma exploração avançada e livre da estrutura dos grafos de conhecimento. O Wizard permite a construção de análises a partir de respostas a diferentes questões, previamente criadas, que permeiam a métrica escolhida.

## 1 Introdução

No mundo atual a tecnologia impera e estamos em constante contato com ela. Nesse contexto, geramos grandes quantidades de dados todos os dias, seja por meio de interações com conteúdos em redes sociais, leitura de notícias *online*, movimentações bancárias por aplicativos ou em compras em diferentes websites de venda, etc. Uma vez que esses dados são gerados, se tornam uma fonte rica de informações.

Sabe-se que é possível analisar dados de diferentes formas e com diferentes ferramentas, de acordo com a natureza daquela informação e de qual o propósito com a sua análise [3]. Por exemplo, dada uma certa quantidade de pessoas que fazem investimentos e seus respectivos montantes, deseja-se saber quantas dessas pessoas fizeram um investimento maior do que  $X$  reais. Diante desse cenário, você pode realizar tal análise com diferentes ferramentas e construir formas de tornar esse estudo acessível para outras pessoas, seja por meio de uma planilha no Excel, um gráfico em uma ferramenta matemática mais robusta ou até mesmo por meio de algum algoritmo.

Conforme os dados e as análises se tornam mais complexas, as ferramentas disponíveis no mercado se tornam mais escassas. É justamente nesse contexto que esse trabalho se enquadra. Imagine um cenário de visualização das empresas clientes de uma empresa, ao

---

\*Instituto de Computação, Universidade Estadual de Campinas, 13081-970 Campinas, SP.

longo dos anos, em que se deseja construir uma forma de verificar como são as relações de venda e compra entre essas empresas e como elas se manifestam. Essa análise é complexa e abre espaço para se discutir formas de se analisar os dados, considerando a melhor forma de representá-los e outras várias perguntas para atingir o objetivo proposto.

Grafos se apresentam como uma forma de estruturar os dados, de um determinado momento no tempo. No contexto do exemplo, empresas seriam representados como vértices e as suas conexões, como arestas. Ademais, haveriam várias versões desse grafo, representando essas conexões em diferentes momentos no tempo. Dessa forma, não há somente a questão de como representar os grafos em sua unidade, mas igualmente o caso de representar as suas mudanças após cada iteração, aumentando a complexidade do problema da visualização.

Esse campo de estudo é muito recente e ainda não há disponível no mercado uma forma de visualização amigável consolidada sobre grafos dinâmicos, mesmo que existam diferentes abordagens que analisam diferentes características dos mesmos. A ferramenta de software *TKGEvolViewer* [1] lida diretamente com esse problema e busca criar formas de visualização de grafos conhecimento dinâmicos.

Grafos de Conhecimentos (*KG*) são compostos por triplas *RDF* (*Resource Description Framework*) [4], na forma de sujeito-predicado-objeto (*i.e.*, conceitos são ligados com relações, como em Empresa A—compra de—Empresa B); os Grafos de Conhecimento Temporais (*TKG*) são compostos por mais de uma iteração de um *KG*. Dessa forma, um *TKG* contém representações das triplas de um determinado domínio em diversos momentos no tempo.

Há diversas aplicações da visualização desses, como descrito por Beck *et al.* [5]: em mapeamento de interações entre proteínas na biologia, transações econômicas no mundo financeiro ou até mesmo performance de jogadores nos esportes. Rossanez *et al.* [6] aplicaram o conhecimento de visualização de grafos em diversas etapas da literatura biomédica sobre doenças degenerativas, através da geração de diversas ontologias — modelo de dados que representa um conjunto de conceitos e os relacionamentos entre eles.

Este trabalho visa efetuar o design e a implementação de novas funcionalidades no *TKGEvolViewer* [1]. Visamos melhorar a compreensão sobre a análise de *TKGs*. Em nossa abordagem, desenvolvemos um Wizard, como um guia em uma interface simples e progressiva com componentes que guiam o fluxo de uso por meio de perguntas e opções de escolha. Para usuários avançados, habilitamos um modo exploratório da ferramenta e componentes mais complexos.

O resultado obtido foi a criação de duas novas formas de visualização no *TKGEvolViewer*: o modo **Wizard**, com a escolha de diferentes cenários de métricas e suas respectivas perguntas, por meio de uma interface simples em um guia progressivo para o usuário iniciante; e o modo **Exploratório**, que possibilita o usuário a estudar os grafos ao longo dos anos por conta própria e exploratória, visualizando a estrutura dos conceitos com o uso do mouse, através de *zoom-in*, *zoom-out* e translação na representação; além do etiquetamento e realce dos vértices, bem como de suas arestas.

Em ambas as visualizações, a ferramenta de software construída permite a visualização interativa de um *TKG*, com seus *KGs* separados espacialmente em ordem cronológica — em formato de linha do tempo, representando seus nós (sujeitos e objetos) com círculos e

suas arestas (predicados) com segmentos de retas e setas.

O restante deste documento está organizado da seguinte maneira: A seção 2 apresenta os refinamentos desenvolvidos no *TKGEvolViewer*. A Seção 3 apresenta uma avaliação de usabilidade conduzida e seus resultados obtidos; A seção 4 apresenta a conclusão deste trabalho.

## **2 Construção do Wizard no *TKGEvolViewer***

Neste estudo, a ferramenta *TKGEvolViewer* [1] foi refinada para se tornar ainda mais intuitiva e prática de representação gráfica e interativa. As suas telas iniciais foram refatoradas para conterem mais informações sobre o funcionamento do software, bem como novos botões de ajuda foram incluídos ao longo da jornada do usuário. Em particular, concebemos um novo modo Wizard, para tornar a ferramenta mais inclusiva e de fácil compreensão.

### **2.1 Construção de um novo Wizard**

A primeira parte foi motivada pela necessidade de se ter uma forma guiada e em um formato mais introdutório, para pessoas iniciantes na ferramenta. Grafos de conhecimento e suas visualizações ainda são novos e não existem muitas ferramentas de visualização disponíveis, as pessoas usuárias precisam ser introduzidas ao assunto e receberem auxílio guiado durante a sua jornada.

Listados critérios necessários para guiar novos usuários no uso inicial: a explicação da ferramenta precisa estar de forma clara e fácil ao abrir a aplicação; a análise e visualização dos dados inseridos deve ser guiado por meio de perguntas (passar a ideia de progressão); devem ser fornecidas informações de auxílio ao longo desse passo a passo; e as pessoas precisam conseguir alterar as informações preenchidas a qualquer momento.

Com base nesses requisitos, exploramos a ferramenta Figma [7] para esboçar a prototipação de telas e modificações à estrutura atual de design da ferramenta. Para facilitar esse processo, todas as telas já existentes foram projetadas no Figma.

Com as telas antigas projetadas pensamos em cenários de análise pré-existentes, que deveriam ser escolhidos pelas pessoas ao entrar no modo Wizard e começar o fluxo de uso da aplicação. Cada um desses cenários refletiria em uma métrica ao final do passo a passo e início das visualizações. Os cenários escolhidos foram:

- Verificar a existência de certo conceito;
- Quais são os conceitos mais importantes;
- Quais são os "n" primeiros conceitos mais importantes;
- Qual a variação de importância de um determinado conceito;
- Quais são os conceitos que importam para um determinado conceito base;
- Quais são os "n" conceitos mais importantes para um determinado conceito;

- Qual a variação que um conceito contribui para a importância de um conceito base;
- Quais são os conceitos que fazem parte da maioria das triplas;
- Quais são os conceitos que representam *subjects* que fazem parte da maioria das triplas;
- Quais são os conceitos que representam *objects* que fazem parte da maioria das triplas;

Mapeamos toda a lista de critérios a serem escolhidos pelas pessoas, de forma a auxiliar na construção dos resultados a serem obtidos. Isso resultou em uma lista de cenários de análise dentro do novo formato do Wizard. Em seguida, apresentamos esses cenários:

### 2.1.1 Cenários de análise e métricas finais

---

**Cenário:** Um certo conceito - **Dado de entrada:** o conceito;

**Critérios de seleção pós escolha do cenário:** Anos disponíveis; Anos não disponíveis; Anos que o conceito começa a aparecer; Anos que o conceito para de aparecer;

**Métrica:** Existência do conceito nos grafos de conhecimento

---

**Cenário:** Conceitos mais importantes

**Critérios de seleção pós escolha do cenário:** Os conceitos mais importantes ao longo do tempo; O conceito mais importante em um determinado ano - **Dado de entrada:** ano; O conceito mais importante em janelas de anos - **Dado de entrada:** conjunto de anos;

**Métrica:** Top-ranked Harmonic (subject, predicate, object, subclassOf, member) centrality;

---

**Cenário:** Um conjunto de conceitos mais importantes (por exemplo, os primeiros 10 / acima de um limite "valor de importância");

**Critérios de seleção pós escolha do cenário:** Ao longo do tempo; em um determinado ano - **Dado de entrada:** ano; em janelas de anos - **Dado de entrada:** conjunto de anos;

**Métrica:** N top-ranked Harmonic (subject, predicate, object, subclassOf, member) centrality;

---

**Cenário:** Determinar a variação de importância de um determinado conceito - **Dado de entrada:** o conceito;

**Critérios de seleção pós escolha do cenário:** Ao longo do tempo; em um determinado ano - **Dado de entrada:** ano; em janelas de anos - **Dado de entrada:** conjunto de anos;

**Métrica:** Harmonic (subject, predicate, object, subclassOf, member) centrality do conceito escolhido;

---

**Cenário:** Os conceitos que contribuem para a importância de um conceito base - **Dado de entrada:** o conceito;

**Critérios de seleção pós escolha do cenário:** Ao longo do tempo; em um determinado ano - **Dado de entrada:** ano; em janelas de anos - **Dado de entrada:** conjunto de anos;

**Métrica:** All Pagerank (subclassOf, member) (conceito escolhido);

---

**Cenário:** Um conjunto de conceitos que contribuem para a importância de um conceito base (por exemplo, primeiros 10 / limite) - **Dado de entrada:** conceito, n, limite;

**Critérios de seleção pós escolha do cenário:** Ao longo do tempo; em um determinado ano - **Dado de entrada:** ano; em janelas de anos - **Dado de entrada:** conjunto de anos;

**Métrica:** Top N Pagerank (subclassOf, member) (conceito base escolhido);

---

**Cenário:** A variação na qual um conceito contribui para a importância de um conceito base (por exemplo, grafo  $i$  grafo de conhecimento temporal) - **Dado de entrada:** conceito base, conceito;

**Critérios de seleção pós escolha do cenário:** Ao longo do tempo; em janelas de anos - **Dado de entrada:** conjunto de anos;

**Métrica:** Pagerank (subclassOf, member) (concept);

---

**Cenário:** Os conceitos que fazem parte da maior parte de triplas;

**Critérios de seleção pós escolha do cenário:** Ao longo do tempo; em um determinado ano - **Dado de entrada:** ano; em janelas de anos - **Dado de entrada:** conjunto de anos;

**Métrica:** Degree centrality;

---

**Cenário:** Os conceitos que representam *sujeitos* que fazem parte da maioria das triplas;

**Crítérios de seleção pós escolha do cenário:** Ao longo do tempo; em um determinado ano - **Dado de entrada:** ano; em janelas de anos - **Dado de entrada:** conjunto de anos;

**Métrica:** Out-Degree centrality;

---

**Cenário:** Os conceitos que representam *objetos* que fazem parte da maioria das triplas

**Crítérios de seleção pós escolha do cenário:** Ao longo do tempo; em um determinado ano - **Dado de entrada:** ano; em janelas de anos - **Dado de entrada:** conjunto de anos;

**Métrica:** In-Degree centrality

---

Com esses cenários mapeados, projetamos as novas telas, sempre levando em conta que o novo modo deveria servir de guia para as pessoas usuárias, então deveria possuir componentes que tornassem esse modo acessível e instrutivo ao longo das diferentes escolhas que fossem tomadas. Durante esse processo de construção, foram feitas algumas iterações, consultando pessoas que seriam público alvo da ferramenta sobre os novos componentes e telas; e como elas se sentiam frente àquelas modificações.

Ao final do processo, chegamos em telas finais que possibilitam que as pessoas selecionem o cenário desejado, preencham os critérios de análise correspondente ao cenário, revisem as informações preenchidas e, assim, possam visualizar graficamente os resultados. Todas essas telas podem ser observadas na seção de apêndices ao final do documento.

### 2.1.2 Desenvolvimento das Telas e do fluxo de interação do modo Wizard

Com as telas implementadas no Figma, foram desenvolvidas as novas telas do modo Wizard em código, sempre se baseando no que foi projetado, utilizando as tecnologias que estão listadas na seção "Tecnologias para implementação" deste documento.

Explicando sobre o novo fluxo de interação com o modo Wizard dentro da ferramenta, a pessoa usuária começa sua jornada escolhendo esse modo de visualização, após o carregamento dos dados do arquivo escolhido, conforme a figura abaixo mostra:

## Concepção e implementação de um *wizard* para análise de grafos de conhecimentos temporais<sup>7</sup>

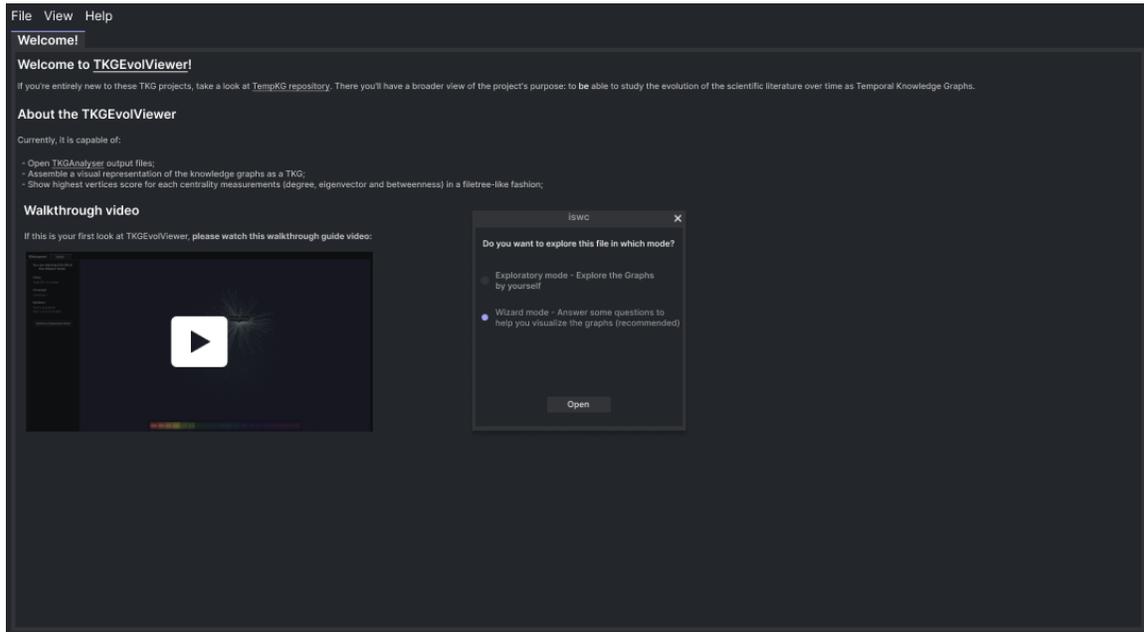


Figura 1: Após a finalização do processamento dos dados, é exibida uma tela de escolha do modo de visualização e análise dos dados, dando a opção da pessoa usuária escolher entre o modo Wizard (guiado) e o modo Exploratório (por conta própria).

Após seleccionar o modo, a pessoa será levada para uma nova tela com todas os cenários de análise dos grafos de conhecimento:

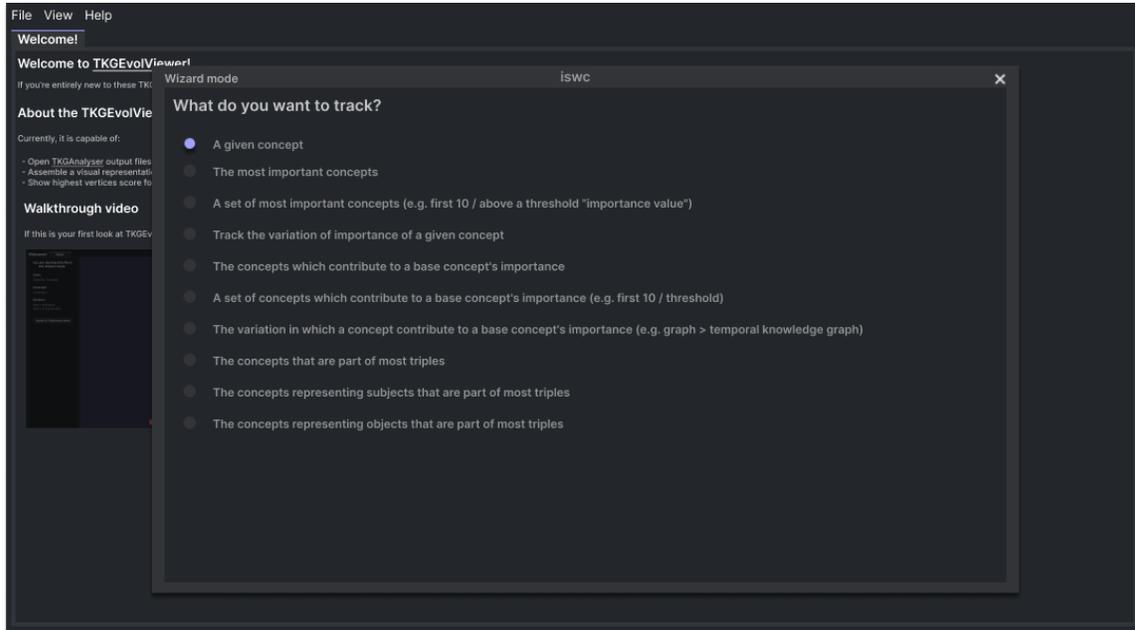


Figura 2: Caso a pessoa escolha o modo Wizard, ela é apresentada à uma tela de escolha do cenário de análise dos grafos de conhecimento gerados. Esse cenário ajudará a guiar a pessoa para a análise e visualização desejada.

## Concepção e implementação de um wizard para análise de grafos de conhecimentos temporais<sup>9</sup>

Dessa forma, ela poderá escolher o cenário de análise que guiará a posterior visualização dos dados carregados. Caso a pessoa tenha dificuldade, pode-se notar que nessa tela, e nas demais, sempre haverá um botão de ajuda para mediar as ações dentro do sistema. Logo que esse cenário é selecionado, a pessoa será levada para uma nova janela com um passo a passo de escolhas de alguns critérios de análise, que permeiam o cenário escolhido (figura 3).

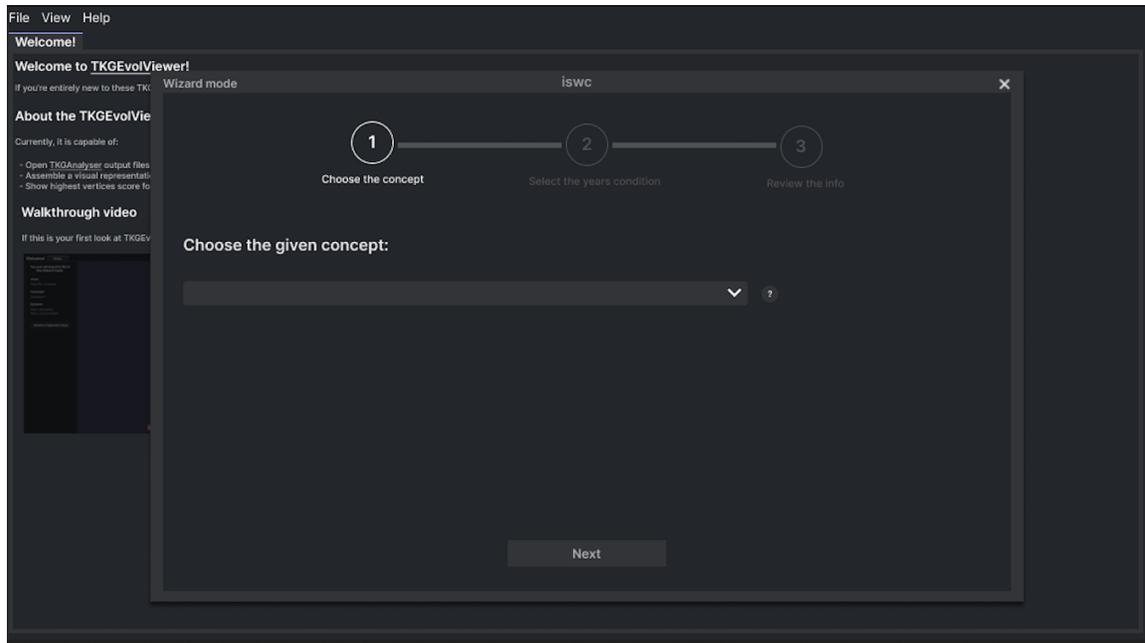


Figura 3: Depois de selecionado o cenário de análise, a pessoa será levada para uma janela com um passo a passo de escolha de diferentes critérios, que permeiam o cenário escolhido e que auxiliarão na visualização e análise também.

A pessoa pode acompanhar o progresso dela ao longo desse passo a passo através da barra de progressão no topo da janela e pode voltar ao passo desejado sempre que precisar, basta clicar no passo desejado com o mouse e/ou nos botões de *Back* ou *Next*. Todas essas etapas possuem o objetivo de guiar a pessoa usuária sobre quais informações podem ser extraídas e guiar a análise e visualização a ser gerada.

Uma vez que a pessoa concluiu todos esses passos de escolha de critérios, é apresentado para ela um resumo de tudo que foi selecionado até então, o cenário e todos os critérios 4:

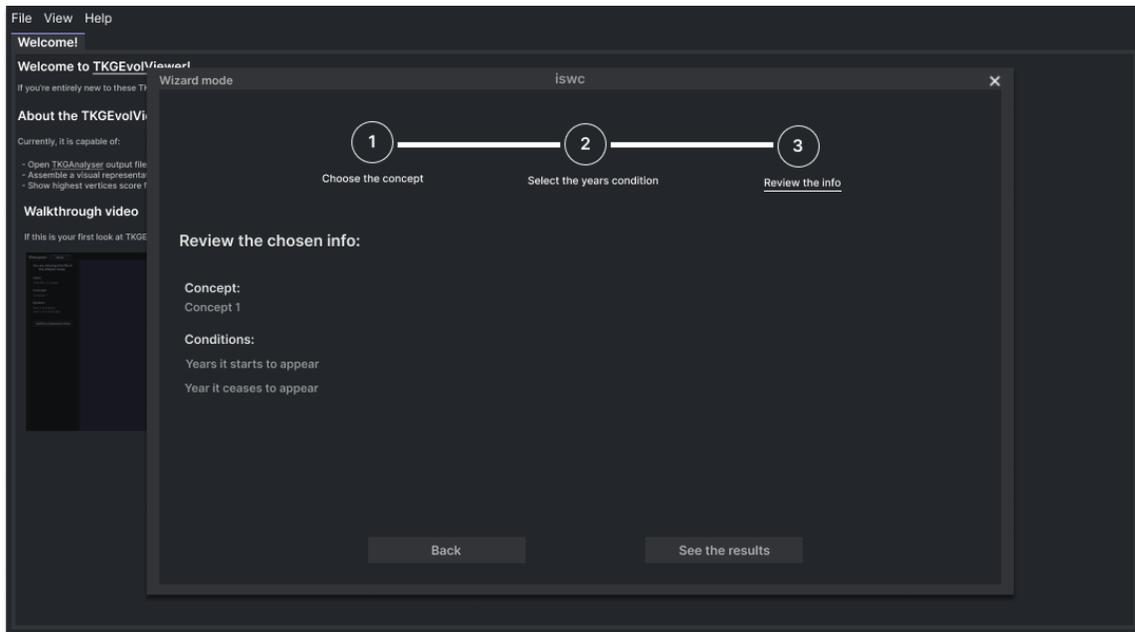


Figura 4: Finalizadas essas escolhas, é exibida uma tela com um resumo de tudo que foi selecionado até então: métricas e critérios associados. Caso tudo esteja correto, basta clicar em "See the results" para continuar e apresentar a representação gráfica dos dados selecionados.

Após fazer essa revisão e confirmar que essas são as condições escolhidas por ela, basta clicar em *See the results*. Isso fará com que a janela atual seja fechada e a de carregamento apareça novamente, visto que novos processamentos precisarão ser feitos e que as condições selecionadas sejam satisfeitas.

Finalizado esse carregamento, o TKGEvolViewer irá apresentar os resultados e uma visão parecida com a figura 5 será apresentada à pessoa usuária, mostrando a métrica guia selecionada em destaque, assim como as suas conexões ao longo da estrutura espacial do grafo, para todas os critérios e anos de análise escolhidos.

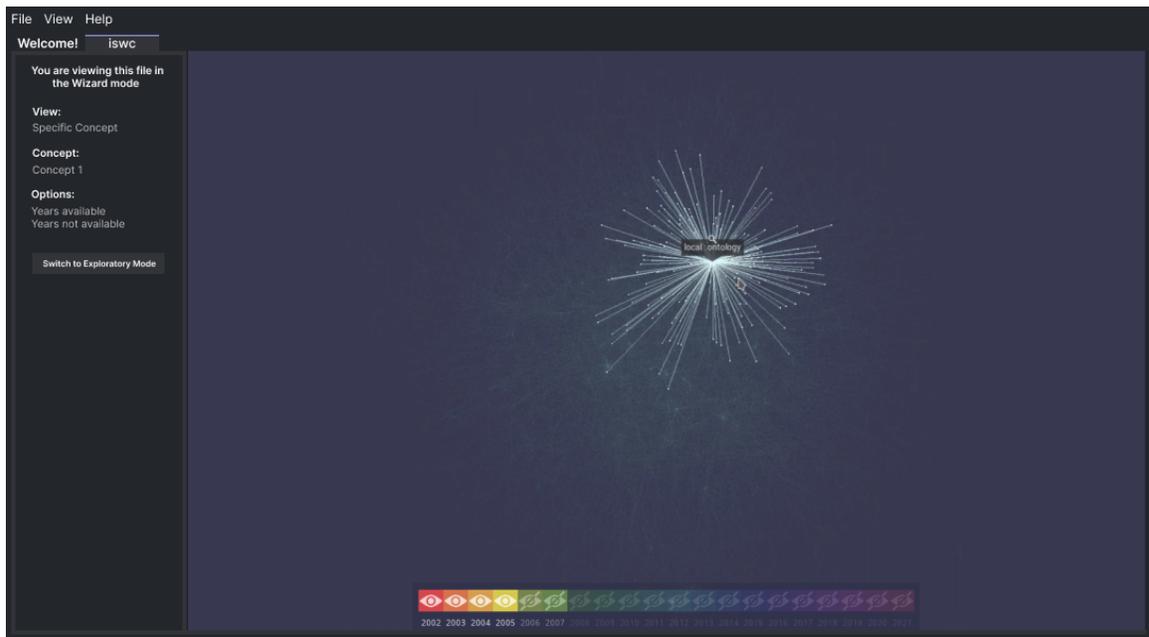


Figura 5: Exemplo de execução do TKGEvolViewer, com a métrica guia sendo um conceito específico (*A given concept*): ‘ontology’ em todas as iterações possíveis, utilizando-se o conjunto de dados ‘iswc’. À esquerda da tela, observa-se o menu lateral comentado anteriormente para auxílio nas edições do Wizard e/ou troca de modo de visualização. Por fim, abaixo do grafo apresentado, é possível visualizar um linha do tempo para relacionar os *timestamps* com as cores empregadas em cada iteração deste, bem como ícones indicando a visibilidade atual de cada uma das iterações.

Dessa maneira, a representação gráfica da base de dados estará pronta para ser explorada livremente. Porém, caso a pessoa possua interesse em modificar algum dos critérios ou métrica selecionada, existe um menu lateral que permite a edição de tudo que foi, previamente, preenchido na ferramenta.

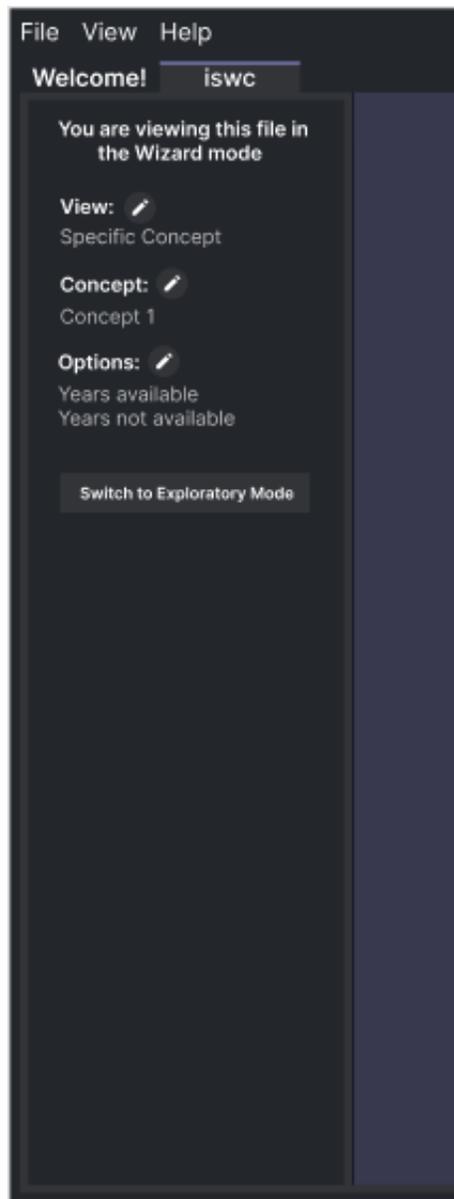


Figura 6: Menu lateral que permite a visualização das informações escolhidas durante o passo a passo do Wizard e troca entre modos.

## 2.2 Construção do modo exploratório

O modo Exploratório foi motivado pela necessidade de existir uma forma rápida e prática de acessar a representação gráfica dos dados inseridos; e explorar por conta própria as métricas e relacionamentos dos TKGs ao longo dos anos. O modo exploratório serve para as pessoas mais experientes com o uso da ferramenta.

Para a idealização dessa parte, projetamos em como seria uma boa experiência para acessar esse modo e trocar entre ele e o Wizard, caso necessário. Para isso, desenhamos uma forma fácil de selecionar, após a inserção dos dados, se a visualização seria feita no modo Wizard ou no modo exploratório.

Como mencionado na seção anterior, foi projetada uma nova janela, mostrada novamente abaixo, que é exibida logo após a finalização do carregamento e processamento dos dados, que permite com que a pessoa consiga escolher o modo de visualização desejado.

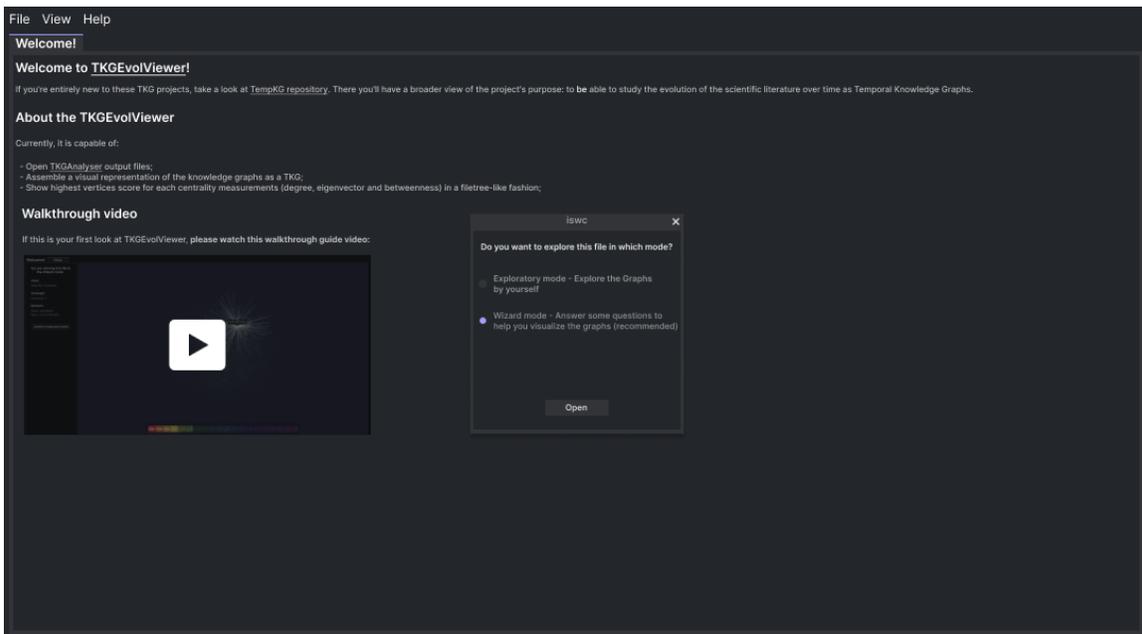


Figura 7: Após a finalização do processamento, é exibida uma tela de escolha do modo de visualização e análise dos dados, dando a opção da pessoa usuária escolher entre o modo Wizard (guiado) e o modo Exploratório (por conta própria).

Em seguida, desenhamos o caso em que a pessoa está dentro de um dos dois modos e deseja trocar para o outro. Para que a experiência de visualização gráfica não precisasse ser fechada, projetamos um botão no menu lateral, que permite a troca rápida entre os modos, e um texto acima do botão que exibe o modo de visualização atual (cf. Figura 6). Ambos foram iterados algumas vezes para obtermos a percepção e a opinião de partes interessadas.



Figura 8: Menu lateral que permite a visualização das informações escolhidas durante o passo a passo do Wizard e troca entre modos.

## 2.3 Melhorias gerais no software

Na tela de início os textos foram modificados e um vídeo demonstrativo dos fluxos de uso gerais foi adicionado (cf. Figura 9); botões de ajuda foram adicionados ao longo do passo a passo e escolhas do novo Wizard(cf. Figura 2); o menu lateral, durante a visualização gráfica dos dados, foi refatorado para conter as informações escolhidas no modo Wizard, se necessário, e possibilitar a troca entre os dois modos de uso (Wizard e Exploratório), assim como mencionado anteriormente (cf. Figura 6).

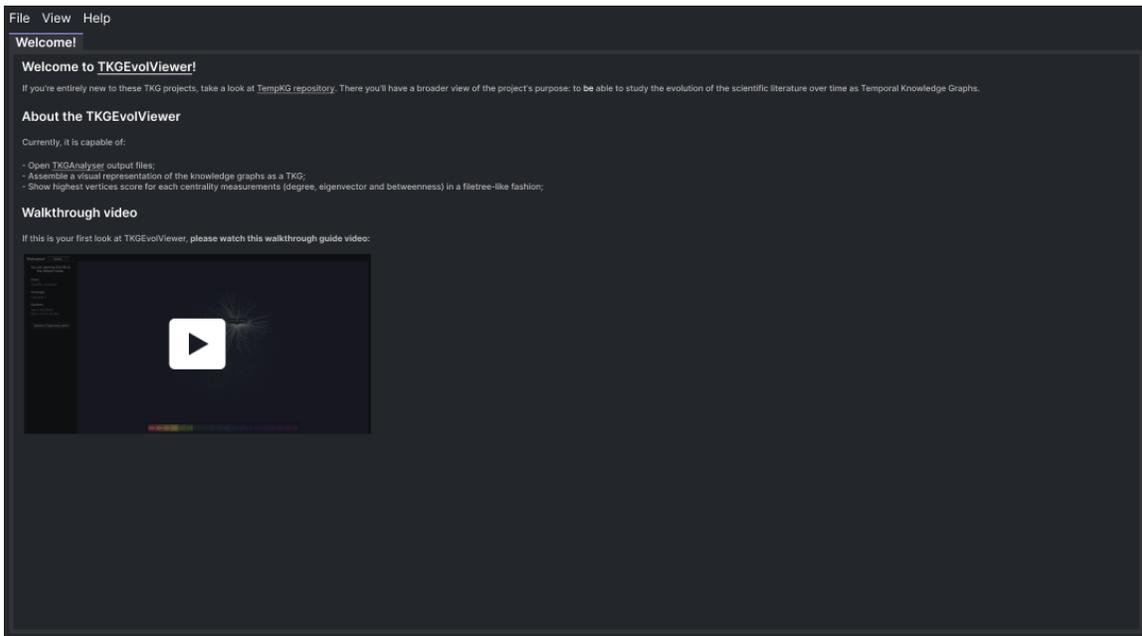


Figura 9: Tela de boas-vindas. É a primeira tela apresentada para a pessoa usuária da ferramenta, exibindo um vídeo demonstrativo, alguns textos de contextualização do projeto e dos programas relacionados. Também é visível uma barra de opções, com os botões *File*, *View* e *Help*.

## 2.4 Tecnologias para implementação

A ferramenta *TKGEvolViewer* [1] foi desenvolvida com o objetivo de ser performática, visto que os grafos analisados podem atingir um alto número de vértices e arestas. Por isso, a tecnologia utilizada para a construção e adição de novas funcionalidades à ferramenta foi a *Godot Game Engine*[8], bem como bibliotecas adicionais para lidar com o processamento dos dados e realização das operações, sendo elas: a *igraph* [9], a *boost* [10] e a *annoy* [11].

As modificações desenvolvidas nesse trabalho utilizaram essa mesma tecnologia base. Sobre a *game engine* em si, ela possui alta capacidade de gerar elementos gráficos com pouca complexidade de implementação; possui uma interface gráfica prática para auxiliar no desenvolvimento; e também tem uma linguagem similar ao Python para a construção de novas interfaces. Isso a torna muito intuitiva para pessoas que estão vindo de outras linguagens de programação.

Outra funcionalidade do *Godot*, que facilita a criação de elementos, é a estruturação em árvore dos módulos desenvolvidos, o que facilita na organização do projeto e manutenção do mesmo, já que muitos elementos podem ser reutilizados ao longo dos cenários gráficos construídos. Vale comentar que o *Godot* possui documentações relevantes, com grande contribuição da comunidade por ser *open source*. Permite ainda fácil integração com bibliotecas externas, o que facilitou a construção do *TKGEvolViewer* [1].

Sobre essas bibliotecas externas, mantidas também da primeira versão da ferramenta, foi utilizada a biblioteca em C *igraph* [9], para o posicionamento de vértices dos grafos. A biblioteca de concorrência do *boost* [10], para otimizar o tempo de cálculo das diferentes iterações de cada um dos *TKGs*, visto que ela faz os cálculos paralelamente para cada um dos grafos. E, por fim, a biblioteca *annoy* [11] para a detecção da posição do cursor do mouse em relação aos grafos.

Para a parte de *Design* e construção de interfaces utilizamos a plataforma *Figma* [7], a qual apresenta uma grande quantidade de componentes Web e Mobile nativos para a criação do *design* das telas. Os protótipos de alta fidelidade permitem e facilitam a criação de navegações para simular a experiência real de uma pessoa com as telas construídas. Todas as telas da primeira versão do *TKGEvolViewer* [1] foram criadas no Figma, juntamente com todas as novas elaboradas neste trabalho, previamente à codificação das mesmas.

### 3 Avaliação

Conduzimos uma avaliação de forma a averiguar a que ponto as novas funcionalidades cumpriram o seu papel de tornar o *TKGEvolViewer* mais prático, intuitivo e simples para pessoas de diferentes experiências de uso e conhecimento sobre o assunto.

A avaliação deveria considerar o uso, tanto do modo Wizard (guiado) quanto do modo Exploratório, de preferência, separadamente e com diferentes pessoas avaliando cada um deles. A forma de avaliação deve ser por meio de pequenas missões que a pessoa deve concluir, utilizando o método *think aloud* [12] para descrever o passo a passo de suas ações.

#### 3.1 Conjunto de dados

Os dados utilizados nas análises dos grafos de conhecimento e realização das avaliações foram gerados a partir do processamento de textos da literatura científica biomédica, através de duas ferramentas: o *KGen* [13] e o *TKGAnalyser* [14].

O *KGen* [13] é responsável por, através de um texto em linguagem natural, fornecer informações para a construção de um Grafo de Conhecimento. Isso é feito através de um pré-processamento, extração de triplas e ligação a elementos de ontologias do texto fornecido, produzindo, assim, os seguintes tipos de elementos:

- *Statements*: Afirmações enumeradas, compartimentalizando as triplas do texto, representadas como:

```
local:s2002.82.1 a rdf:Statement ;
    rdfs:label "the two areas meet today" .
```

- *Classes*: Definição dos sujeitos e objetos da triplas, representadas como:

```
local:two_area a rdf:Class ;  
  rdfs:label "two area" .
```

- *Mapped Relations*: Ligação de classes locais a conceitos já existentes em ontologias, representados como:

```
local:world_wide_web owl:sameAs cso:world_wide_web .
```

- *Relations*: Ligação de classes e afirmações através de mapeamentos locais e já existentes em ontologias, representados como:

```
local:two_area rdfs:subClassOf local:area .  
local:area_meet_today rdfs:subClassOf local:today .  
local:area_meet_today rdfs:member local:area_meet .  
local:s2002.82.1 rdf:subject local:two_area .  
local:s2002.82.1 local:AM-TMP local:today .  
local:s2002.82.1 rdf:predicate local:meet .
```

O *TKGAnalyser* [14] é responsável por analisar as métricas dos vértices criados pelo *KGen*, ordenando os conceitos locais de interesse em várias listas e gerando um arquivo de acordo com a seguinte estrutura JSON:

```
[  
  {  
    "year": "2002",  
    "tkg": "https://raw.githubusercontent.com/.../2002/merged_kg.ttl",  
    "metrics": [  
      {  
        "metric": "degree",  
        "concepts": [  
          { "http://local/local.owl#semantic_web": 0.02578361981799798 },  
          { "http://local/local.owl#ontology": 0.008088978766430738 }, ... ]  
        },  
      {  
        "metric": "indegree",  
        "concepts": [ ... ]  
      }, ...  
    ]  
  }
```

```

  },
  {
    "year": "2003",
    "tkg": "https://raw.githubusercontent.com/.../2003/merged_kg.ttl",
    "metrics": [ ... ]
  }, ...
]

```

Para a análise dentro da ferramenta construída (*TKGEvolViewer*), é necessária somente o arquivo gerado pelo *TKGAnalyser*, pois, para cada iteração do processamento dos textos da literatura biomédica, há uma referência online para o arquivo gerado pelo *KGen*.

A base utilizada contém mais de 26 mil *Statements*, cerca de 89 mil *Classes*, aproximadamente 2 mil *Mapped Relations* e mais de 230 mil *Relations*, além de contar com 12 métricas de centralidade disponíveis para análise, todos entre os anos de 2002 e 2021.

### 3.2 Materiais e protocolos

Criamos um ambiente propício para o uso da ferramenta e execução da avaliação. Os seguintes protocolos e materiais foram utilizados e disponibilizados durante a realização da avaliação:

- Computador com a ferramenta de software em execução e pelo menos um arquivo com os TKGs salvos localmente;
- Roteiro para condução do teste para a pessoa intermediadora;
- Um tempo máximo de duração do teste de 30 minutos foi definido;
- Um ambiente tranquilo e calmo foi escolhido para a condução do teste;
- Recrutamento e Seleção das pessoas participantes.
- Foram definidos dias e horários com as pessoas avaliadas para realização do teste;

### 3.3 Condução da avaliação

Projetamos três etapas distintas levando em consideração as novas funcionalidades adicionadas na ferramenta:

**1) Entendimento inicial e conhecimento da ferramenta** Aqui desejava-se avaliar o quão claro/compreensível está o propósito da ferramenta o que se pode fazer dentro dela, bem como quais são os primeiros passos a seguir.

**2) Modo Guiado (Wizard)** Esta etapa tinha por objetivo introduzir a pessoa a ferramenta, bem como guiar a visualização dos grafos, auxiliando na criação de análises e entendimento dos dados selecionados. Portanto, a ideia dessa etapa foi verificar se o Wizard estava funcionando como um guia e sendo claro e descritivo o suficiente para pessoas que

não possuem muito conhecimento e/ou que possuem, mas que querem algum tipo de ajuda para realizar a análise desejada.

**3) Modo exploratório** Nessa etapa a proposta foi avaliar o quão útil/relevante é a ferramenta ter esse modo exploratório, bem como identificar se o mesmo dá uma visão geral dos grafos analisados, sem nenhum tipo de auxílio inicial.

Com essas três etapas definidas, cada uma das pessoas, participantes do teste ficou responsável por receber uma missão, que representa o objetivo descrito na etapa; então concluir sozinha a tarefa, mas descrevendo o processo de raciocínio dela e comentando sobre as decisões e dificuldades que enfrentar.

Para finalizar o processo de condução, falando da pessoa intermediadora, essa ficou responsável por anotar todo o processo de interação com a ferramenta e a passagem pelas etapas e missões feitas pelas pessoas participantes. O intermediador ficou responsável por orientar sobre algum ponto que a pessoa tenha dúvida e que não interfira nos resultados do teste.

### 3.4 Participantes

Uma vez definidos os protocolos e materiais, dois participantes, atualmente Engenheiros de Software e formados em Engenharia da computação, foram recrutados para a realização dos testes, pois possuíam conhecimento sobre grafos e são da área da computação.

Ambos foram recrutados por meio de troca de mensagens, explicando sobre a ferramenta, trabalho e dando contexto sobre o possível teste que aconteceria com essas pessoas. Alguns artigos sobre TKGs foram compartilhados para ajudar na contextualização.

Após essa conversa inicial, os dois participantes aceitaram participar do teste e foram combinadas as datas de execução do mesmo. Antes da realização do teste, termos de consentimento foram preenchidos por ambos. O primeiro participante ficou responsável por testar o modo Wizard; o segundo ficou responsável por testar o modo Exploratório; ambos participantes avaliariam a explicação inicial e entendimento da ferramenta ao abri-la (etapa 1).

### 3.5 Resultados e discussão

**Primeira avaliação.** Na primeira avaliação o primeiro participante fez o uso do modo Wizard. Seu objetivo foi abrir um arquivo salvo no computador disponibilizado, visualizá-lo no modo Wizard, selecionar um cenário de sua preferência, preencher o passo a passo e explorar os TKGs gerados. Essa avaliação demorou cerca de 23 minutos. Consideramos as seguintes lições aprendidas.

O participante ficou com dúvida sobre como o arquivo que estava salvo na máquina local havia sido gerado e o que os dados que estavam presentes nele significavam. Nesse momento, a pessoa intermediadora (Pardal) explicou sobre o *KGen* e *TKGAnalyser* novamente e como aqueles dados haviam sido gerados e também os princípios que representavam os dados. Após essa explicação, o participante soube o que esperar para continuar a realização da etapa.

Em seguida, o participante assistiu o vídeo e elogiou o fato dele existir dentro da tela inicial da ferramenta. Contou que sabia agora o que esperar, quais as funcionalidades presentes e o passo a passo que deveria ser realizado. Em seguida, selecionou o arquivo corretamente; comentou que era parecido com a seleção padrão em outros software e avançou.

Após o período de *loading* dos dados, soube selecionar o modo Wizard corretamente, mas comentou que poderia haver alguma explicação do motivo desses dois modos existirem, pensando em pessoas iniciantes. Passado o *feedback*, foi aberta a tela de cenários e o participante clicou no botão de ajuda para entender mais sobre o que estava sendo pedido. Lida a ajuda, comentou que poderia ter algum tipo de explicação sobre cada um dos cenários; por exemplo, mais botões de ajuda espalhados ao longo deles. Conseguiu selecionar o cenário de “Um determinado conceito” e avançou.

Em seguida, teve a oportunidade de explorar o passo a passo de seleção de critérios e elogiou a barra de progresso, movimentação entre passos e botões de ajuda presentes. Comentou que não sentiu problemas para avançar e conseguiu chegar até a tela de revisão das informações e confirmar as seleções para visualizar os resultados.

Nesse momento teve a tela de carregamento e, em seguida, a representação gráfica dos grafos apareceu. O participante aproveitou esse espaço para explorar e entender sobre mais sobre os dados, elogiando a fluidez da movimentação, a interação com os grafos e as cores marcantes. Após passar alguns minutos explorando, disse que estava satisfeito com a análise e soube fechar a aba atual de visualização, finalizando o teste.

Como *feedbacks* finais, comentou que gostou da ferramenta proposta e achou ela intuitiva; em especial na interação com os grafos, pois foi onde conseguiu mais entender os dados e compreender o propósito real da ferramenta.

**Segunda avaliação.** O participante fez o uso do modo Exploratório, em que a pessoa intermediadora objetivou selecionar um arquivo já salvo localmente, no computador disponibilizado; visualizá-lo no modo exploratório; explorar os grafos gerados e trocar para o modo Wizard ao final do processo. Essa avaliação demorou cerca de 14 minutos e gerou as seguintes lições aprendidas.

Diferentemente da primeira avaliação, nesse já foi explicado e dado o contexto sobre o que os dados a serem analisados representavam e como haviam sido gerados, evitando a dúvida, anteriormente, citada. Ao entrar na ferramenta, o participante leu os textos e já abriu o vídeo explicativo.

Após terminar de assisti-lo, elogiou a existência e ficou impressionado com o que a ferramenta poderia oferecer. Em seguida, soube abrir a janela de seleção de arquivos e abrir o arquivo corretamente. Uma vez que o carregamento dos dados foi finalizado, a tela de escolha do modo de visualização foi aberta e o participante leu com calma para selecionar a opção correta. Nesse momento, comentou que os textos descritivos, para cada um dos modos, estavam claros e que ajudaram na escolha correta do Exploratório.

Com o modo selecionado, o participante foi levado para a tela com a representação gráfica dos dados diretamente e pôde explorar a ferramenta e os dados em exibição. Passou algum tempo visualizando e interagindo com cada uma das arestas considerando nós e ano de iteração dos dados. Assim que sentiu que havia explorado o que desejava, decidiu trocar o modo de visualização para o Wizard, assim como o protocolo da avaliação demandava.

Para esse fim, primeiramente tentou clicar no botão de "File" no topo da ferramenta e então depois que não encontrou a opção, que viu que havia um botão no menu lateral para a troca do modo. Comentou aqui que a informação poderia estar presente nos dois lugares e/ou que o botão/texto poderiam ser maiores para dar mais destaque ao modo selecionado; e como trocá-lo, se necessário. Finalizou a avaliação trocando de modo.

Como comentários finais, o participante elogiou a fluidez da visualização gráfica dos dados e a interação com os grafos, bem como a rapidez para entrar no modo exploratório e exibir os dados. Comentou que faria a alteração mencionada sobre a alternância entre os modos. Comentou igualmente que poderia existir mais links e explicações sobre grafos de conhecimento na tela de início da ferramenta, visto que esse era um dos principais focos do software.

Conduzidos as avaliações e os resultados apresentados, notamos que nosso objetivo inicial foi atingido, de tornar a ferramenta mais prática, simples e que seja adaptável para pessoas de diferentes experiências de uso. Isso porque em ambas avaliações conduzidas os participantes conseguiram concluir os objetivos descritos, sem tirar dúvidas sobre o processo intermediário para atingi-los.

Relataram que o vídeo demonstrativo da página inicial ajudou a entender como a ferramenta funcionava e quais funcionalidades estariam presentes para uso, bem como facilita o primeiro uso por pessoas que nunca haviam interagido com a ferramenta antes. Outra funcionalidade elogiada foi a presença de textos informativos ao longo dos dois modos de informação, os quais ajudam a entender o que está sendo requisitado naquele momento e como a pessoa deveria prosseguir e interagir.

## 4 Conclusão

A visualização de grafos de conhecimento junto à extração de métricas e visualização dos grafos é um tópico de pesquisa que ainda demanda estudos, principalmente no que se refere à diferentes formas de visualização dessas informações e interação com as mesmas. Neste trabalho desenvolvemos refinamentos na ferramenta *TKGEvolViewer* visando facilitar a condução das análises na ferramenta. A nova funcionalidade **Wizard** permite às pessoas escolherem diferentes métricas e serem conduzidas ao longo das perguntas pela aplicação. Igualmente, nossa solução permite que pessoas mais experientes possam fazer uso do modo exploratório e analisar os grafos. Acreditamos que mudanças ainda podem ser desenvolvidas para melhorar a experiência do usuário no *TKGEvolViewer*. Trabalhos futuros visam desenvolver novas versões explorando novas formas de visualização 3D dos TKGs.

## Referências

- [1] DE SOUZA, Eduardo Moreira Freitas; DOS REIS, Julio. TKGEvolViewer. **TKGEvolViewer**, 2021. Disponível em: <<https://www.ic.unicamp.br/~reltech/PFG/2021/PFG-21-34.pdf>>. Acesso em: Março de 2022.
- [2] Ehrlinger L, Wöß W. Towards a definition of knowledge graphs. **12th International conference on semantic systems (SEMANTiCS2016)**. 2016.
- [3] NEO, Benedict. The Six Types of Data Analysis. **Towards Data Science**. 2020. Disponível em: <<https://towardsdatascience.com/the-six-types-of-data-analysis-75517ba7ea61>>. Acesso em: Junho de 2022.
- [4] O. Lassila, R. R. Swick, W. Wide, W. Consortium, Resource description framework (rdf) model and syntax specification (1998).
- [5] BECK, Fabian; BURCH, Michael; DIEHL, Stephan; WEISKOPF, Daniel. A Taxonomy and Survey of Dynamic Graph Visualization. **COMPUTER GRAPHICS Forum**. 2019. v. Volume 36. 2017. p. 133-159.
- [6] ROSSANEZ, Anderson; DOS REIS, Julio; TORRES, Ricardo Da Silva. Representing Scientific Literature Evolution via Temporal Knowledge Graphs. In proceedings of the 6th Workshop on Managing the Evolution and Preservation of the Data Web (MEPDaW), co-located with the 19th International SemanticWeb Conference (ISWC). **MEPDaW**. 2020. p. 33-42.
- [7] Figma UI Design Tool. **Figma**. Disponível em: <<https://www.figma.com/ui-design-tool/>>. Acesso em: Abril de 2022.
- [8] Godot Engine. **Godot**. Versão 3.4. Página inicial. C++. Disponível em: <<https://godotengine.org/>>. Acesso em: Fevereiro de 2022.
- [9] Igraph - The network analysis package. **igraph**. Versão 0.9. Página inicial. C. Disponível em: <<https://igraph.org/>>. Acesso em: Março de 2022.
- [10] Boost C++ libraries. **boost**. Versão 1.77. Página inicial. C++. Disponível em: <<http://boost.org/>>. Acesso em: Março de 2022.
- [11] Annoy (Approximate Nearest Neighbors Oh Yeah). **annoy**. Versão 1.17. Github, 2022. C++, Python, Lua, Go, C. Disponível em: <<https://github.com/spotify/annoy>>. Acesso em: Março de 2022.
- [12] Wikipedia. **Think Aloud Protocol**. Disponível em: <<https://en.wikipedia.org/wiki/Think-aloud-protocol>>. Acesso em: Maio de 2022.
- [13] ROSSANEZ, Anderson. **Kgen**. Github, 2021. Python. Disponível em: <<https://github.com/rossanez/KGen>>. Acesso em: Março de 2022.

- [14] ROSSANEZ, Anderson. **TKGAnalyser**. Gitlab, 2021. Python. Disponível em:  
<<https://gitlab.ic.unicamp.br/ra124136/TKGAnalyser>>. Acesso em: Março de 2022.