

QAnubis - Uma Ferramenta Web de Apoio ao Processo de Análise Qualitativa

Giliardi Schmidt

Universidade Federal do Pampa
Alegrete, Brasil
gili.schmidt@hotmail.com

Ildevana Poltronieri Rodrigues
Universidade Federal do Pampa
Alegrete, Brasil
ildevanarodrigues@unipampa.edu.br

Elder de Macedo Rodrigues

Universidade Federal do Pampa
Alegrete, Brasil
elder.rodrigues@unipampa.edu.br

Fábio Paulo Basso

Universidade Federal do Pampa
Alegrete, Brasil
fabio.basso@unipampa.edu.br

Maicon Bernardino

Universidade Federal do Pampa
Alegrete, Brasil
bernardino@unipampa.edu.br

Lucas Pieva

Instituto Federal de Educação Ciência
e Tecnologia Farroupilha
Santo Ângelo, Brasil
lucas.pieva@iffarroupilha.edu.br

RESUMO

A principal atividade da pesquisa qualitativa é a análise de dados, a qual envolve um processo de coleta e análise de dados narrativos, incluindo textos, fotos, elementos audiovisuais e diversos formatos de arquivos digitais. Diferentes abordagens e metodologias estão disponíveis para a investigação de dados qualitativos, com maior ênfase na codificação, em que artefatos e seus conteúdos são categorizados sob um conjunto finito de categorias. Assim como em outras áreas, são realizados esforços para desenvolver e aprimorar ferramentas computacionais conhecidas como Software de Análise Qualitativa Assistida por Computador, que visam aumentar a eficiência e a eficácia do processo de análise ao fornecer meios para auxiliar na identificação, organização, interpretação, exploração e integração dos dados. Com o intuito de contribuir para o campo da análise qualitativa, este trabalho apresenta uma ferramenta web de código aberto chamada QAnubis, voltada para a realização de codificação em arquivos PDF, preservando seu conteúdo e formatação originais. Para conceber essa aplicação, o presente trabalho define os requisitos, as decisões de projeto e os detalhes de implementação, além de apresentar uma ilustração das funcionalidades da ferramenta. Por fim, é apresentada uma avaliação empírica, que buscou a opinião de especialistas sobre o software desenvolvido. O QAnubis foi considerado relevante na área devido à sua natureza de código aberto e à inclusão de funcionalidades essenciais de análise.

Link do vídeo: <https://doi.org/10.5281/zenodo.15483877>

PALAVRAS-CHAVE

Análise qualitativa, ferramentas, CAQDAS, QAnubis

1 Introdução

A pesquisa qualitativa tem uma natureza exploratória, que busca compreender significados, ideias, motivações e qualidades em dados narrativos, como textos, fotos, elementos audiovisuais e diferentes formatos de arquivos digitais. Embora tenha surgido nas ciências sociais, como na antropologia e na história, essa abordagem tem encontrado aplicação em áreas como Engenharia de Software (ES). Na ES, a pesquisa qualitativa é utilizada para compreender fenômenos como dinâmicas de equipes, comunicação entre desenvolvedores e análise de feedback de usuários. Com base nesses exemplos, destaca-se sua crescente relevância para estudos acadêmicos e aplicações na indústria, onde pode auxiliar em análises estratégicas e avaliativas.

Diferente de uma pesquisa quantitativa, que utiliza números para sintetizar as conclusões, a pesquisa qualitativa tem como apelo o uso de palavras, com as quais pode-se gerar explicações e resultados mais descriptivos, tornando-se muitas vezes mais convincentes. Apesar de suas diferenças, ambos os métodos são complementares, permitindo uma visão mais completa ao combinar abordagens quantitativas e qualitativas, especialmente em contextos como Engenharia de Software, onde dados técnicos e humanos se entrelaçam [8].

Sendo uma abordagem multidisciplinar, a pesquisa qualitativa oferece uma variedade de metodologias e técnicas para a análise de dados [4]. Um elemento central nesse processo é o *coding*, que permite categorizar os artefatos e seus conteúdos em conjuntos organizados de códigos que representam significados subjacentes. Essa técnica é amplamente empregada na ES, particularmente em estudos que investigam padrões em interações humanas ou organizacionais, como análise de requisitos, *feedback* de usuários e documentação técnica. O *coding* possibilita estruturar *insights* qualitativos de maneira que sejam acessíveis e úteis para a tomada de decisão.

Assim como em outras áreas, há uma tendência de modernização e automação de tarefas. Isto ocorre também na análise qualitativa, onde diversas ferramentas CAQDAS (*Computer-Assisted Qualitative Data Analysis Software*) estão disponíveis para serem utilizadas por pesquisadores. Estes *softwares* visam auxiliar o pesquisador com funcionalidades para identificar, organizar, explorar, interpretar e integrar os dados em análise [12]. Novas funcionalidades e *softwares* são propostas frequentemente, visando oferecer suporte a diferentes extensões de arquivos e técnicas de análise. Tendo em vista os benefícios da utilização de CAQDAS, é importante que pesquisadores e empresas contribuam com a democratização do acesso a ferramentas *open source* que auxiliem no processo de análise qualitativa.

O objetivo deste trabalho é apresentar a análise, especificação, projeto e implementação de uma ferramenta que apoie a análise qualitativa. Ela tem como propósito oferecer o conjunto básico de funcionalidades que apoiam as diferentes etapas de uma análise qualitativa, tais como a importação de artefatos, codificação dos dados, escrita de anotações analíticas e geração/visualização dos resultados.

Este trabalho está organizado da seguinte forma: os principais conceitos relacionados ao estudo são apresentados na Seção 2. Na

Seção 3 são apresentados os requisitos, decisões de projeto, arquitetura, detalhes de implementação, como também as funcionalidades da ferramenta proposta. Os resultados de uma avaliação preliminar são apresentados na Seção 4. Por fim, na Seção 5 são apresentadas as considerações finais e trabalhos futuros.

2 Referencial teórico

A pesquisa qualitativa é composta por um conjunto de técnicas interpretativas, sendo difícil definir o que a compõem com exatidão, por não possuir uma teoria ou paradigma nitidamente próprio [1]. Essa abordagem incorpora métodos e ferramentas de diversas áreas do conhecimento, como ciências sociais, saúde e tecnologia [9]. Na Engenharia de Software, a pesquisa qualitativa tem sido usada para explorar aspectos subjetivos e complexos de fenômenos relacionados a equipes, processos e produtos. Geralmente, o processo envolve três etapas principais: (i) coleta de materiais empíricos alinhados ao tópico investigado; (ii) análise detalhada dos dados; e (iii) produção de resultados interpretativos que oferecem *insights* significativos [1]. A principal atividade do processo de pesquisa qualitativa é a análise qualitativa de dados [4]. Ela consiste na classificação e interpretação de materiais textuais e visuais, visando elaborar afirmações sobre as informações explícitas e implícitas do material em análise e o que está representado nele. Ao se realizar uma análise qualitativa, busca-se descrever algum fenômeno, comparar artefatos, buscar explicações ou desenvolver teorias sobre o material em análise [4]. Em geral, pode-se classificar em duas categorias as abordagens para analisar dados qualitativamente, sendo que muitas vezes ambas são utilizadas em conjunto [4]. A primeira classificação é orientada a expandir o material da análise ao criar interpretações na forma de anotações textuais, com base de observações realizadas sobre os dados. Já na segunda categorização, busca-se reduzir a complexidade dos dados da análise através da realização do *coding*.

2.1 O coding na análise qualitativa

O *coding* é uma técnica essencial da análise qualitativa, onde se cria um conjunto estruturado de códigos para agrupar elementos de dados com base em significados compartilhados ou ideias centrais. Estes elementos podem ser textos, imagens, áudios, etc., ou qualquer conjunto de dados interligados, ou não. O objetivo destes códigos não é categorizar o conjunto de palavras ou símbolos, mas sim o seu significado [8]. Estes códigos podem ser representados por siglas, palavras, frases curtas, cores, etc., que simbolicamente atribuem ou capturam a essência dos dados, sendo que cada código pode ser aplicado a mais de um dado [10].

A execução do *coding* é iterativa por natureza, frequentemente exigindo múltiplas rodadas de refinamento. Cada iteração permite gerenciar, filtrar e ajustar os códigos para melhor representar os padrões emergentes nos dados [10].

Conforme apresentado na Figura 1, o *coding* vai além da criação de códigos. Ao decorrer da análise, o pesquisador pode identificar e criar categorias (e subcategorias), de modo a agrupar esses códigos, com base em padrões e semelhanças. Estas categorias e códigos podem, então, ser utilizadas para criar temas/conceitos e teorias/afirmações [10]. No contexto da Engenharia de Software, o *coding* tem sido empregado para identificar tendências em *feedback*

de usuários, compreender processos colaborativos em equipes e explorar padrões em dados não estruturados.

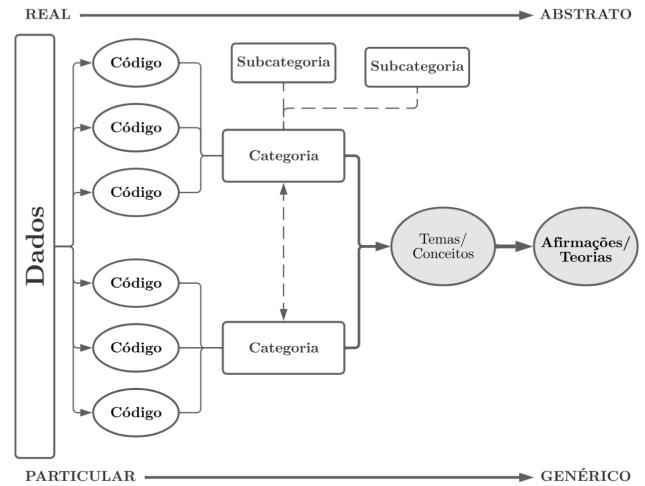


Figura 1: Simplificação do processo de transformar códigos em teorias [10].

2.2 Ferramentas computadorizadas de análise qualitativa de dados

Desde sua origem na década de 1980 [4], as ferramentas CAQDAS têm sido utilizadas para apoiar diferentes aspectos do processo de análise qualitativa. No entanto, não há uma definição universalmente aceita para definir o que constitui uma CAQDAS, visto que esse termo engloba ferramentas com variados objetivos e conjuntos de funcionalidades que de alguma forma, apoiam a execução de análises qualitativas. Assim, como este apoio pode se manifestar de diferentes maneiras, ferramentas com diferentes objetivos e conjuntos de funcionalidades podem ser classificadas como CAQDAS, porém todas possuem como características prover meios para analisar, visualizar ou codificar dados (realizar o *coding*) [12].

Para ser classificada como uma CAQDAS, a ferramenta não precisa abranger todas as possíveis funcionalidades. É suficiente que ela forneça um conjunto básico que permita ao pesquisador realizar análises qualitativas significativas. Algumas das funcionalidades frequentemente encontradas em CAQDAS incluem [12]:

- **Planejamento e gerenciamento de projeto:** uma CAQDAS pode ser vista como um repositório do seu trabalho. Todas as informações relacionadas e dados a serem analisados podem ser centralizados em um único lugar;
- **Escrita de anotações analíticas:** criação e desenvolvimento de anotações que ajudam o pesquisador a manter registros sobre suas descobertas e pensamentos. Este conjunto de funcionalidades pode ser, por exemplo, um diário de projeto ou painel com comentários;
- **Leitura, marcação e comentários nos dados:** a CAQDAS pode oferecer meios para destacar e comentar sobre informações importantes, com o intuito de facilitar a exploração dos dados;

- **Busca de informações:** para facilitar ainda mais a exploração dos dados, pode-se disponibilizar meios de buscar informações, como, por exemplo, uma funcionalidade de buscar por uma determinada palavra ou frase em um documento de texto;
- **Coding:** o *coding* é uma das principais abordagens para analisar dados qualitativamente. Ao ser realizado em uma CAQDAS, o pesquisador aplica aos artefatos em análise (ou os seus conteúdos), um conjunto finito de códigos;
- **Visualização de segmentos codificados:** após ou durante o *coding*, pode ser interessante para o pesquisador visualizar um resumo dos dados codificados, como, por exemplo, em uma lista que reúne todos os segmentos relacionados à um determinado código.

2.3 Trabalhos relacionados

Uma Revisão Sistemática da Literatura (RSL) foi conduzida para identificar e caracterizar ferramentas CAQDAS, abrangendo a análise de bibliotecas digitais e literatura cinza¹. O levantamento inicial resultou em 2551 estudos e 43 ferramentas. A RSL revelou que a área de análise qualitativa é amplamente explorada, tanto na academia, com foco no desenvolvimento e aprimoramento de ferramentas, quanto na indústria, onde ferramentas avançadas oferecem funcionalidades robustas para apoiar análises qualitativas.

Após a aplicação de critérios de inclusão, exclusão e qualidade, 49 estudos foram selecionados e classificados em seis categorias principais: (i) 27 trabalhos utilizaram CAQDAS para analisar dados qualitativos em diversos formatos e áreas de conhecimento, destacando-se artefatos relacionados a entrevistas, como textos, vídeos e áudios; (ii) nove estudos propuseram novas ferramentas voltadas à análise de arquivos textuais, *tweets* e conteúdos de plataformas específicas; (iii) seis compararam características ou funcionalidades de CAQDAS e métodos manuais; (iv) três propuseram funcionalidades adicionais para facilitar visualização de dados, codificação de arquivos textuais e relações entre códigos; (v) dois sugeriram melhorias para aprimorar a experiência do usuário, especialmente para iniciantes, ressaltando a relevância de documentações claras; (vi) um estudo apresentou um exemplo de uso da ferramenta *Inquire Tool* e outro propôs um *guideline* para o desenvolvimento de ferramentas para análise de arquivos de áudio relacionados a entrevistas.

3 Ferramenta web de apoio ao processo de análise qualitativa

Nas seções a seguir serão apresentados os requisitos, decisões de projeto, tecnologias e apresentação das principais funcionalidades da ferramenta.

3.1 Requisitos da ferramenta

Esta seção apresenta os Requisitos Funcionais (RF) definidos para a ferramenta, com base nos resultados obtidos com a realização da RSL e nas definições apresentadas no referencial teórico. Estes requisitos foram definidos tendo em mente abranger as principais etapas de um processo de análise qualitativa, oferecendo funcionalidades

¹Uma versão detalhada da RSL pode ser encontrada em no repositório Zenodo publicada como Relatório Técnico: <https://doi.org/10.5281/zenodo.15484345>.

para a realização do *coding* e anotação de informações analíticas, no formato de diários de projeto.

- **RF1. Cadastro de usuário:** A ferramenta deve permitir que o usuário cadastre-se no sistema e edite suas informações pessoais. Isso é necessário para manter a rastreabilidade entre o usuário e seus projetos cadastrados na ferramenta;
- **RF2. Autenticação:** A ferramenta deve permitir ao usuário autenticar no sistema. A autenticação garante ao usuário a permissão para acessar somente seus projetos, impedindo que pessoas não autorizadas obtenham informações confidenciais ou alterem o trabalho de outros;
- **RF3. Recuperação de senha:** A ferramenta deve permitir que o usuário já cadastrado no sistema possa recuperar (alterar) sua senha;
- **RF4. Gerenciamento de projetos:** A ferramenta deve permitir que o usuário gerencie seus projetos. O usuário deve conseguir listar, criar, editar e deletar projetos. Cada projeto representa uma análise qualitativa e é composto por diversos artefatos a serem analisados (arquivos PDF), códigos (*codes*), diários de projeto e citações (*quotes*), *trechos de texto codificados*;
- **RF5. Gerenciamento de códigos:** A ferramenta deve permitir que o usuário gerencie os códigos relacionados a cada projeto. O usuário deve conseguir listar, criar, editar e deletar códigos;
- **RF6. Gerenciamento de arquivos PDF:** A ferramenta deve permitir que o usuário gerencie os arquivos PDF relacionados a cada projeto. O usuário deve ser conseguir listar, adicionar, editar dados e deletar estes arquivos;
- **RF7. Coding em arquivos PDF:** A ferramenta deve permitir que o usuário faça o *coding* sobre estes arquivos. O *coding* consiste na seleção de trechos de texto e vinculação de códigos aos mesmos. Além disso, o usuário deve poder adicionar comentários a estes trechos codificados;
- **RF8. Gerenciamento de diários de projeto:** A ferramenta deve permitir que o usuário gerencie diários de projetos. O usuário deve conseguir listar, criar, editar e deletar diários de projeto;
- **RF9. Geração de relatórios:** A ferramenta deve permitir que o usuário gere relatórios, que devem centralizar e sumarizar os resultados da análise qualitativa na forma de gráficos e tabelas.

3.2 Decisões de projeto

Esta seção apresenta as Decisões de Projeto (DP) tomadas para o desenvolvimento da QAnubis.

- **DP1. Suporte para arquivos PDF:** Análises qualitativas podem ser realizadas em diferentes artefatos, como, vídeos, fotos, arquivos de áudio e outros documentos digitais. Nisto, identificou-se a oportunidade de implementar uma ferramenta *open-source web* que suporte o *coding* de arquivos com extensão PDF de maneira "completa", visto que nenhum *software* com estas características foi identificado na RSL. Este suporte "completo" refere-se a como o usuário visualiza um arquivo PDF em análise, pois na revisão realizada

somente foram encontradas ferramentas *open-source* que extraem o conteúdo textual dos arquivos PDF e só exibem o texto ao usuário. Já no suporte “completo”, a ferramenta deve exibir todo o conteúdo do PDF, em sua formatação original. Além disso, ao suportar essa extensão, indiretamente a ferramenta dará suporte a outras extensões, tendo em vista que o formato PDF é gratuito, *open-source* e é possível transformar uma grande variedade de extensões de arquivos para PDF.

- **DP2. Licença *open-source*:** Conforme apresentado na DP1., identificou-se a oportunidade de implementar uma ferramenta *open-source web* que suporte o *coding* de arquivos com extensão PDF. Além disso, o código-fonte estará disponível para qualquer pessoa visualizar e modificar, trazendo como vantagens a evolução e manutenção colaborativa.
- **DP3. A solução deve ser um sistema *web*:** Conforme apresentado nas outras decisões de projeto, para contribuir com algo novo na área de CAQDAS, decidiu-se desenvolver uma ferramenta *web* para análise de PDF. Para isso, escolheu-se utilizar uma arquitetura cliente-servidor, permitindo que a utilização do sistema seja feita por navegadores *web*, independente do sistema operacional que o usuário estiver utilizando. Por este motivo, a linguagem de programação escolhida foi o PHP², para o lado do servidor, e a linguagem Javascript³ para o lado do cliente. O PHP foi escolhido porque é *open-source* e amplamente utilizado em sistemas de pequeno e grande porte, o que tende a garantir que seja constantemente atualizado com melhorias de desempenho e segurança, além não consumir muitos recursos (memória e poder de processamento) do servidor e ser de fácil aprendizagem. Já o Javascript, é a principal linguagem de programação utilizada para tornar páginas *web* interativas, pois é interpretada pela grande maioria dos navegadores *web*. Para facilitar a evolução e manutenção do sistema, foram escolhidos os frameworks Laravel⁴, para o lado do servidor, e Bootstrap⁵, para o lado do cliente. Estes frameworks, como muitos outros, trazem benefícios, tais como reuso de funcionalidades, segurança, legibilidade e documentação.
- **DP4. Língua inglesa:** tendo em vista abranger um público maior, os textos da ferramenta devem estar localizados para a língua inglesa, visando que esta linguagem é considerada universal na ciência.

3.3 Tecnologias

Para desenvolver a aplicação, utilizou-se a plataforma PHPStorm⁶ para a escrita do código-fonte, o MySQL⁷ como banco de dados e o GitHub⁸ para o versionamento do projeto. Foram utilizadas as seguintes versões para os frameworks Laravel (v8.15) e Bootstrap (v4.6). Para o PHP, se utilizou a versão 7.3, sendo esta a mínima necessária para a versão do Laravel empregada. A utilização de dependências é inevitável, pois elas aceleram o desenvolvimento do

software. Por isso, para facilitar o versionamento do código-fonte, fez-se o uso do Composer⁹ e NPM¹⁰, para o gerenciamento de dependências PHP e Javascript, respectivamente.

3.4 A ferramenta QAnubis

Esta seção detalha a ferramenta QAnubis e suas funcionalidades principais, desenvolvidas para apoiar o processo de análise qualitativa de dados. O código-fonte, os requisitos e as instruções para implementação estão disponíveis em um repositório público¹¹ no GitHub, licenciado sob a Apache 2.0¹². A fim de demonstrar o uso prático da ferramenta, um vídeo foi produzido, apresentando desde o fluxo inicial de uso até os recursos mais avançados: <https://doi.org/10.5281/zenodo.15483877>.

O acesso ao sistema QAnubis é iniciado por meio de uma página de autenticação, onde o usuário deve fornecer um *e-mail* e senha válidos. Adicionalmente, o sistema permite a recuperação de senhas, enviando um link único e temporário para redefinição, caso o envio de *e-mail* esteja habilitado. A criação de contas exige o preenchimento de informações básicas, como nome, *e-mail* e senha. Após o cadastro, o usuário deve confirmar seu *e-mail* para ativar a conta, garantindo segurança e autenticidade.

Após autenticar-se no sistema, é exibida ao usuário a listagem dos seus projetos e informações quantitativas sobre os códigos, documentos, citações e diários que os compõem, podendo filtrar estes projetos por nome ou ordená-los por nome ou última alteração. Além disso, nesta mesma tela, o usuário pode cadastrar, editar informações básicas ou remover permanentemente um projeto. Ao criar ou editar um projeto, o usuário informa um nome, descrição e cor. Para realizar o *logout*, como também acessar outras partes do sistema, o usuário pode utilizar o menu *dropdown*, presente na barra superior. Após autenticado, o usuário também pode editar os seus dados de cadastro e senha, por uma página acessível por este *dropdown*.

Ao acessar um projeto, é exibido ao usuário uma página (Figura 2), contendo um botão para explorar citações e 5 abas, relacionadas ao gerenciamento de documentos, códigos, diários de projeto, relatórios e informações do projeto.

Na **aba de documentos** (*documents*), selecionada por padrão, é exibida a listagem dos PDF anexados aos projetos, com botões para abrir, editar o nome, visualizar citações relacionadas e remover o arquivo. O usuário pode realizar uma busca textual para filtrar os dados desta listagem, como também ordenar as linhas conforme os dados das colunas, por ordem crescente/decrescente. Estas ações sob a listagem também estão disponíveis para códigos e diários. Além disso, há um botão para realizar o *upload* de um novo PDF.

A **aba de códigos** (*codes*) consiste na exibição da listagem de códigos criados pelo usuário, com botões de edição, remoção e exibição de citações relacionadas, como também um botão para adição. Estes códigos são representados por um nome, cor, descrição abreviada, descrição completa e código pai (utilizado para criar uma hierarquia de códigos). Há duas maneiras de visualizar esses códigos: por uma tabela, semelhante aos documentos, e uma árvore hierárquica.

²PHP: <https://www.php.net/>

³Javascript: <https://www.javascript.com/>

⁴Laravel: <https://laravel.com/>

⁵Bootstrap: <https://getbootstrap.com/>

⁶PHPStorm: <https://jetbrains.com/pt-br/phpstorm/>

⁷MySQL: <https://www.mysql.com/>

⁸GitHub: <https://github.com/>

⁹Composer: <https://getcomposer.org/>

¹⁰NPM: <https://www.npmjs.com/>

¹¹Código-fonte da QAnubis: <https://github.com/gilischmidt/qanubis>

¹²Licença de uso do código-fonte: <https://www.apache.org/licenses/LICENSE-2.0>

Figura 2: Página de edição de um projeto.

The screenshot shows the 'Document Manager' section of the QAnubis interface. At the top, there are several tabs: 'Test Project' (highlighted), 'Explorar as citações' (highlighted), 'Documentos' (highlighted), 'Códigos', 'Memos', 'Relatórios', 'Project Info', and 'Super Admin'. Below the tabs, there's a search bar and a button to 'Add a document'. A red box highlights the 'Documentos' tab. On the left, a sidebar shows 'Current Document' with 'Texto de Ipsum.com' selected. A red box highlights this selection. The main area displays a table of documents with columns: Type, Name, Description, Quotes, and Added at. The table includes rows for 'Example of use.pdf', 'Lorem Ipsum.pdf', 'Random Lorem Ipsum Text.pdf', 'Texto de Ipsum.com' (which is highlighted with a red box), and 'Uses of Lorem Ipsum.pdf'. To the right of the table are five numbered buttons (1, 2, 3, 4, 5) with icons for opening, editing, viewing, and deleting. At the bottom, there are 'Previous' and 'Next' buttons.

Legenda: Botões: 1 - abrir; 2 - editar; 3 - visualizar citações relacionadas; 4 - deletar.

Já a **aba de diários** de projeto (*memos*), assim como nas apresentadas anteriormente, consiste em uma listagem de diários, com botões para acessar, editar, remover e adicionar um novo diário. A tela de edição de um diário é composta por uma área de texto, onde o usuário pode aplicar diversos estilos de formatação.

Na **aba das informações do projeto** (*project info*), o usuário pode editar os dados básicos do mesmo e o deletar.

Ao acessar um documento, o sistema exibe ao usuário um leitor de PDF, onde se pode navegar entre páginas, realizar buscas por texto e alterar o nível de *zoom*. As citações selecionadas são destacadass com as cores dos códigos aplicados. Na esquerda deste leitor, é exibida a listagem de citações. Este painel pode ser ocultado, caso desejado, para aumentar área do leitor de PDF. Em cada elemento desta listagem, há um botão para rolar o leitor até o local onde o texto codificado está localizado. Em contrapartida, caso o usuário clique em um trecho de texto destacado, no leitor, o sistema rola esta listagem até exibir o elemento da lista relacionado. Além disso, ao deixar o *mouse* sob algum trecho codificado, o sistema exibe ao usuário os códigos relacionados e quantidade de comentários relacionados. Para realizar o *coding*, o usuário seleciona um bloco de texto e clica com o botão esquerdo do *mouse*, para exibir a opção de relacionar códigos ao texto. Com isso, o sistema exibe um *modal*, onde o usuário pode selecionar um ou mais códigos e adicionar comentários. Ao clicar na etiqueta (*badge*) que representa um código, o usuário pode editar as suas informações.

Na última aba da página de edição de um projeto, referente aos **relatórios** (*reports*), o usuário pode gerar gráficos e tabelas que sumarizam os dados do *coding*. Os gráficos podem ser exportados para imagens (png, jpg, svg e PDF) e dados (csv, json, xlsx, html e PDF), com exceção do gráfico hierárquico de códigos, sendo que este somente pode ser exportado para svg. Já as tabelas podem ser exportadas para csv, excel e PDF. Ambas as categorias de exibição de resultados podem ser impressas diretamente do navegador. Há também botões para exportar todas as citações para um arquivo txt, agrupados por códigos ou documentos.

Na aba de relatórios, o usuário pode selecionar quais documentos e códigos deseja utilizar na geração dos resultados. Há também a opção de selecionar a granularidade dos códigos, ou seja, se deseja

exibir somente os códigos selecionados, os códigos e seus filhos diretos ou os códigos e todos os seus filhos (incluindo indiretos), tendo em vista que o sistema permite criar uma hierarquia de códigos. Os gráficos disponíveis são: (i) árvore da hierarquia de códigos, onde o usuário pode mover a árvore, aumentar/diminuir o zoom, clicar em um nó para exibir/ocultar os seus filhos ou clicar com o botão direito do *mouse* para exibir um menu com opções de visualizar citações relacionadas ou editar o código. (ii) matriz de relações entre códigos, que demonstra a sobreposição de códigos, ou seja, citações com mais de um código. Neste gráfico, o usuário pode visualizar as citações ao clicar no cruzamento entre dois códigos; (iii) matriz de códigos por documentos, semelhante à matriz de relações entre códigos; (iv) nuvem de palavras das citações, onde o usuário pode excluir, limitar o número e definir o peso mínimo para que uma palavra seja exibida; (v) nuvem de palavras dos códigos, que demonstra graficamente os códigos com mais citações relacionadas, semelhante ao outro gráfico de nuvem. Já as tabelas disponíveis são: (i) lista de documentos selecionados, onde o usuário pode visualizar a quantidade de códigos e citações relacionadas, como também visualizar estas citações; (ii) lista de códigos, onde o usuário pode visualizar a quantidade de documentos e citações relacionadas e visualizá-los.

As funcionalidades de visualizar/explorar as citações, disponível na barra de menu superior, listagem de documentos, códigos e nas tabelas presentes na aba de relatórios, exibem uma janela, ou *modal*, composto por duas seções: filtros e listagem. Nos filtros, o usuário pode selecionar as citações com base nos documentos e códigos relacionados, como também filtrar pela hierarquia dos códigos (se deseja exibir somente os códigos selecionados, os códigos e seus filhos diretos ou os códigos e todos os seus filhos, incluindo indiretos). Já a listagem é composta pelas citações e seus comentários. Pensando em facilitar a utilização do sistema, o usuário pode editar as citações diretamente desta listagem, podendo alterar os códigos e comentários. Além disso, o usuário pode editar os códigos, ao clicar na etiqueta (*badge*), do código.

4 Avaliação inicial da QAnubis

Para a concepção e execução da avaliação, utilizou-se como embasamento os trabalhos de [2], [3] e [5], que apresentam experiências e guias para o planejamento, condução e análise de dados de entrevistas. Para isso, adotou-se um protocolo inspirado pelas diretrizes e sugestões apresentadas por [6], [7] e [11].

Neste contexto, a avaliação seguiu uma abordagem qualitativa, focada nas experiências e percepções de usuários em potencial. Para explorar a relevância e usabilidade da ferramenta, foram conduzidas entrevistas semiestruturadas com quatro participantes. Um formulário¹³ complementar foi utilizado para caracterizar os participantes em termos de formação acadêmica e experiência prévia com análises qualitativas, fornecendo um contexto para avaliar a ferramenta e identificar possíveis melhorias. A avaliação focou nos seguintes aspectos: relevância da ferramenta, facilidade de uso, adequação do conjunto de funcionalidades às necessidades básicas da análise qualitativa e oportunidades de melhoria.

Os participantes consideraram a QAnubis uma contribuição relevante para a área de análise qualitativa, especialmente por sua

¹³Questionário de Avaliação da QAnubis: <https://doi.org/10.5281/zenodo.1548447>

acessibilidade como por ser uma alternativa *open-source*. A ferramenta foi vista como um MVP promissor para CAQDAS, atendendo às necessidades iniciais de análise baseada em *coding*. O *layout* foi elogiado pela simplicidade e pela organização em abas, que facilitam a navegação entre componentes do sistema. Um dos destaques mencionados foi a possibilidade de acesso via internet, solucionando limitações enfrentadas por usuários de ferramentas *desktop*, como a Atlas.ti, que dificultam a mobilidade e a colaboração em projetos. Melhorias sugeridas incluíram aprimoramentos no suporte à hierarquização de códigos e na flexibilidade das opções de visualização. As sugestões de melhorias foram implementadas e estão atualmente disponibilizadas pela ferramenta.

Além das melhorias já implementadas, foram registradas as seguintes sugestões, consideradas relevantes para o aprimoramento futuro, porém não essenciais para o escopo do MVP: (i) aprimoramento da responsividade da aplicação, visando garantir acesso pleno via dispositivos móveis para visualização dos relatórios das análises; (ii) melhoria na visualização das citações no leitor de PDF, uma vez que, apesar de a lista atual permitir a visualização, o crescimento no número de citações poderá demandar formas mais eficientes para o pesquisador localizar e explorar os dados codificados; (iii) suporte para múltiplos usuários em um mesmo projeto, possibilitando edição colaborativa; (iv) transcrição automática e codificação (*coding*) de arquivos de áudio; (v) inclusão de formulários para que o usuário possa cadastrar informações adicionais relativas à análise, tais como o protocolo utilizado e demais dados pertinentes; (vi) implementação de funcionalidade para criação de *links* entre diários e outros artefatos do projeto (documentos, códigos e citações); (vii) habilitação de ações em massa sobre documentos, códigos e citações, visando facilitar o gerenciamento diante do grande volume potencial desses artefatos; (viii) realização de testes de desempenho e otimização do uso de rede, em razão das características mencionadas anteriormente relacionadas ao elevado volume de artefatos; (ix) adição de metadados aos artefatos, como datas de criação, edição, entre outras informações relevantes.

5 Conclusão

Este trabalho teve como objetivo apresentar a QAnubis, uma ferramenta web open-source voltada ao apoio da análise qualitativa por meio do processo de coding diretamente sobre arquivos em formato PDF, mantendo sua formatação original. A principal motivação para seu desenvolvimento surgiu a partir de uma Revisão da Literatura, que identificou uma variedade de ferramentas CAQDAS existentes, mas nenhuma com essa combinação específica de características: acesso via navegador, código aberto e suporte completo a PDFs preservando a estrutura visual e elementos gráficos.

A ferramenta foi projetada com foco na simplicidade, acessibilidade e viabilidade técnica, oferecendo um conjunto essencial de funcionalidades: gerenciamento de projetos, codificação de trechos em arquivos PDF, hierarquização de códigos, escrita de anotações analíticas e geração de relatórios em múltiplos formatos. Esses recursos são organizados de maneira intuitiva, contribuindo para a adoção por usuários com diferentes níveis de experiência.

A avaliação preliminar, conduzida com quatro usuários com experiência em análise qualitativa, indicou que a QAnubis atende de

forma satisfatória às etapas iniciais do processo de análise, especialmente em cenários com dados documentais. Os participantes destacaram como pontos fortes a interface limpa, a facilidade de uso e a viabilidade de uso colaborativo, dado seu acesso web. A avaliação também forneceu sugestões valiosas que foram prontamente incorporadas, como melhorias na hierarquia de códigos e nas visualizações. Reconhece-se, entretanto, que a ferramenta ainda possui limitações importantes. Entre os pontos identificados estão: a ausência de suporte a outros tipos de artefato (áudio, vídeo, imagens), a falta de integração com métodos avançados de análise (como codificação assistida por IA), e a ausência de suporte colaborativo simultâneo. Essas lacunas são reconhecidas e servirão como base para a evolução da ferramenta.

Conclui-se que a QAnubis representa uma contribuição concreta à comunidade de pesquisadores interessados em análise qualitativa, oferecendo uma solução gratuita, extensível e especialmente útil em contextos educacionais ou institucionais com restrições de acesso a ferramentas proprietárias. Espera-se, nos próximos ciclos de desenvolvimento, ser realizado por um aluno de mestrado, ampliar a cobertura funcional da QAnubis e aprofundar sua avaliação empírica com estudos controlados e comparativos com outras ferramentas do mercado.

DISPONIBILIDADE DE ARTEFATO

O Código fonte da QAnubis pode ser encontrado no repositório: <https://github.com/gilischmidt/qanubis> A demonstração em vídeo encontra-se no repositório Zenodo: <https://doi.org/10.5281/zenodo.15483877>. Para fins de avaliação da ferramenta, foi elaborado um questionário, que está disponível em: <https://doi.org/10.5281/zenodo.15484476>. Adicionalmente, uma versão completa da RSL, publicada como Relatório Técnico, pode ser consultada no seguinte link: <https://doi.org/10.5281/zenodo.15484345>.

Referências

- [1] Norman K Denzin and Yvonna S Lincoln. 2006. *O planejamento da pesquisa qualitativa: teorias e abordagens*. Artmed.
- [2] Barbara DiCicco-Bloom and Benjamin F Crabtree. 2006. The qualitative research interview. *Medical education* 40, 4 (2006), 314–321.
- [3] Kevin Dunn. 2000. Interviewing. (2000).
- [4] Uwe Flick. 2013. *The SAGE handbook of qualitative data analysis*. Sage.
- [5] Siw Elisabeth Hove and Bente Anda. 2005. Experiences from conducting semi-structured interviews in empirical software engineering research. In *11th IEEE International Software Metrics Symposium (METRICS'05)*. IEEE, 10–pp.
- [6] Jyrki Kontio, Johanna Bragge, and Laura Lehtola. 2004. Using the focus group method in software engineering: obtaining practitioner and user experiences. In *International Symposium on Empirical Software Engineering*. 271–280. <https://doi.org/10.1109/ISESE.2004.1334914>
- [7] Antonio José Manzato and Adriana Barbosa Santos. 2012. A elaboração de questionários na pesquisa quantitativa. *Departamento de Ciéncia de Computação e Estatística-IBILCE-UNESP* (2012), 1–17.
- [8] Matthew B Miles and A Michael Huberman. 1994. *Qualitative data analysis: An expanded sourcebook*. sage.
- [9] Jane Ritchie, Jane Lewis, Carol McNaughton Nicholls, Rachel Ormston, et al. 2013. *Qualitative research practice: A guide for social science students and researchers*. sage.
- [10] Johnny Saldaña. 2021. *The coding manual for qualitative researchers*. sage.
- [11] Forrest Shull, Janice Singer, and Dag IK Sjøberg. 2007. *Guide to advanced empirical software engineering*. Springer.
- [12] Christina Silver and Ann Lewins. 2014. *Using Software in Qualitative Research: A Step-by-Step Guide* (2nd ed.). Sage Publications Ltd.