

# DX (Developer Experience): revisão de literatura cinza sobre a perspectiva de profissionais da indústria de 2020 a 2022

Gabriel Rabelo de Melo<sup>1</sup>, Awdren de Lima Fontão<sup>1</sup>

<sup>1</sup>Faculdade de Computação – Universidade Federal de Mato Grosso do Sul (UFMS)  
Caixa Postal 549 – 79.070-900 – Campo Grande – MS – Brasil

`gabriel.rabelo@ufms.br`, `awdren.fontao@ufms.br`

**Abstract.** *Developer Experience (DX) is an emergent term that represents the experience of developers with a software product. During the COVID-19 pandemic, human relations underwent big changes, including the way software developers relate to the artifacts consumption and their creation. Using Grey Literature Review, this research investigates the perception of practitioners about DX in selected publications during the pandemic, from the years 2020 to 2022. We summarize the DX concept based on the analysis of selected texts and theoretical references, and discuss the tools and metrics addressed by practitioners.*

**Resumo.** *Developer Experience (DX) é um termo emergente que representa a experiência em relação a um produto de software, por parte de desenvolvedores. Durante a pandemia do COVID-19, as relações humanas sofreram grandes mudanças, dentre elas a forma como desenvolvedores de software se relacionam com o consumo de artefatos e a criação deles. Utilizando Grey Literature Review, esta pesquisa investiga a percepção de profissionais da indústria sobre DX no recorte de publicações durante a pandemia, para os anos de 2020 a 2022. Sintetizamos um conceito sobre DX com base na análise dos textos selecionados e do referencial teórico, e discutimos as ferramentas e métricas abordadas pelos praticantes.*

## 1. Introdução

Em 2020, haviam mais de 56 milhões de desenvolvedores registrados no *GitHub*, segundo dados do relatório anual da organização, o *Octoverse 2020* [GitHub, 2020]. Esse número representou um crescimento de 16 milhões, comparado ao ano anterior, 2019. Para 2021, o crescimento foi de 17 milhões, cerca de 73 milhões de desenvolvedores no total [GitHub, 2021]. E para 2022, o total de desenvolvedores foi de 94 milhões, cerca de 21 milhões a mais que 2021 [GitHub, 2022].

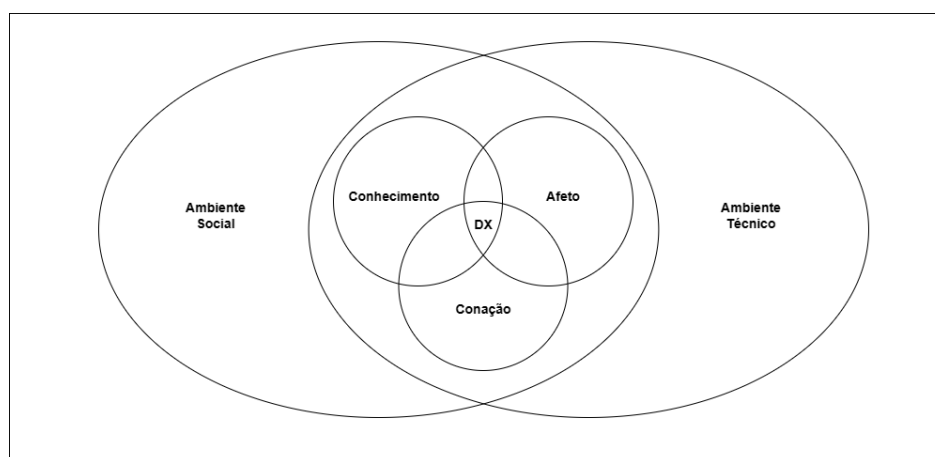
Com base nos dados, nota-se que nos últimos três anos (2020-2022) o crescimento da comunidade global *open-source* no *GitHub* dobrou, saindo de cerca de 40 para 94 milhões de usuários [GitHub, 2020, 2021, 2022]. Há indícios que a pandemia do COVID-19 impactou no aumento da comunidade, conforme relatório de 2020 do *GitHub* [GitHub, 2020]. Os números tendem a serem bem maiores, dado que a comunidade de *software* fechado não é avaliada no relatório. Com essa entrada de novos membros na comunidade de desenvolvedores, novos produtos e inovações surgem. Por consequência, aumentam-se as relações entre desenvolvedores e estes com os novos softwares do mercado [Constantino *et al.*, 2023].

Neto *et al.* (2021) conduziram um estudo sobre os impactos da pandemia no desenvolvimento de *software*. Um dos fatores, sob a perspectiva dos profissionais pesquisados, é o equilíbrio entre trabalho e vida pessoal, assim como a saúde mental. Outros estudos também identificaram esses fatores em relação a instauração de trabalho remoto [Ford *et al.*, 2022; Uddin, Alam e Serebrenik, 2022]. Esse cenário indica que a visão de DX pode estar passando por uma reconfiguração.

Das interações entre a comunidade e produtos surge a necessidade de gerenciar esse relacionamento e potencializá-lo. DevRel (Relações de Desenvolvedores, do inglês *Developer Relations*), é um termo que se refere ao estudo das relações *software*-desenvolvedor [Fontão *et al.*, 2023]. Refere-se também a área dentro de uma organização responsável pela gestão de ativos com os quais ocorrem essas relações entre os atores do ecossistema, promovendo a sinergia entre objetivos organizacionais e expectativas dos desenvolvedores [Fontão *et al.*, 2023].

As interações entre as partes, geram por sua vez, reações empíricas sobre as partes humanas do ecossistema [Dewan, 2015; Manikas, 2016; Fontão *et al.*, 2017; Novielli e Serebrenik, 2019; Fontão *et al.*, 2020]. Essa visão das interações *software*-desenvolvedor sobre a parte humana do processo é definida como *Developer Experience* (DX) [Fagerholm e Münch, 2012]. DX é vista ainda como a estrutura capaz de prover motivação no contexto de engenharia de *software* [Kuusinen *et al.*, 2016].

Fagerholm (2015) cria, com base em seus estudos, um conceito de DX (Figura 2). As relações existentes envolvem tanto o ambiente social como o ambiente técnico. Nessa ponte existe DX como as partes que se convergem do conhecimento, dos aspectos afetivos e da conação. Em relação ao *conhecimento*, abarca todo o conhecimento individual, seja ele de natureza técnica ou não. No espectro do que cabe a *conação*, encontra-se a motivação e as ações realizadas. Já no espectro *afetivo*, as emoções e sentimentos resultantes do convívio no ambiente técnico e social de uma organização.

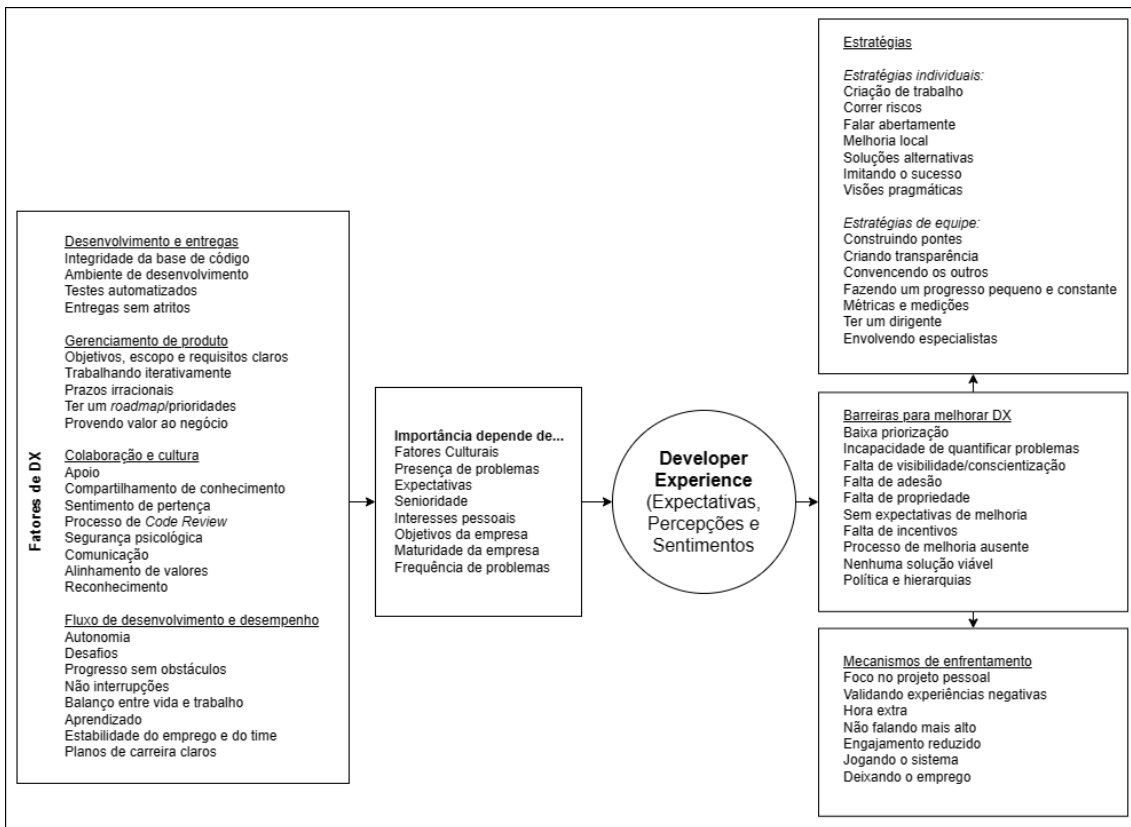


**Figura 2. O Conceito de *Developer Experience* (adaptado de Fagerholm, 2015)**

Greiler *et al.* (2023) conduziram um estudo onde foi possível identificar que DX existe como mecanismo de retenção para as organizações. Retenção essa causada pelos impactos positivos no sentimento de desenvolvedores em continuarem como parte de uma organização. Potrich e Selig (2019) afirmam que a falta de retenção é uma das principais causas de perda de conhecimento para organizações do setor de tecnologia. Conseqüentemente, a falta de retenção ocasiona impactos negativos no negócio.

Além dos impactos internos, os impactos externos também existem, sob a perspectiva de consumo de produtos de *software*. Consumidores insatisfeitos buscarão novos produtos e levarão esse ativo financeiro para outra organização [Zaid e Patwayati, 2021]. Além desse descontentamento afetar a marca negativamente, por meio de comentários sobre a experiência com o produto ou serviço utilizado [Zaid e Patwayati, 2021].

Por sua vez, Greiler *et al.* (2023) apresentam uma proposta de um *framework* para DX (Figura 3) composto por dimensões da mente: *expectativas*, *percepções* e *sentimentos*. Quando comparado com a proposta de Fagerholm (2015), DX, para Greiler *et al.* (2023), é explorada no que o primeiro estudo aborda sobre aspectos afetivos. A pesquisa foi realizada pelo processo investigativo com profissionais da indústria. Evidenciou-se o contexto em que estão envolvidos, para identificar fatores que impactam e as barreiras para a promoção de uma boa DX.



**Figura 3. Framework Acionável de Developer Experience (adaptado de Greiler et al., 2023)**

Nylund (2020) realizou uma pesquisa de revisão de literatura multivocal (MLR, *Multivocal Literature Review*) e afirma que DX é um termo emergente, constatado por meio da quantidade de publicações realizadas em diversos meios de comunicação sobre o tema. Ele evidenciou os objetos e entidades estudados em relação a DX. Assim como os métodos de estudos empregados, os fatores de melhora ou piora de DX, os contextos dos estudos realizados e as definições dadas.

A partir do contexto apresentado, assim como dos trabalhos já realizados, a presente pesquisa é motivada a averiguar a visão dos profissionais da indústria. Ela foi realizada por meio da condução de GLR (*Grey Literature Review*, Revisão de Literatura

Cinza) [Garousi *et al.*, 2019], como forma de prover um melhor entendimento DX. Espera-se que ela possa auxiliar pesquisadores e profissionais da indústria para promoção de estudos e práticas para melhoria de DX.

Foi possível obter como resultados deste estudo: (a) a caracterização de DX em meio a cadeia de desenvolvimento de *software*; (b) uma visão complementar das definições acadêmicas frente a encontrada; (c) assim como, a identificação de ferramentas e métricas utilizadas pela indústria no contexto de DX. Os resultados caracterizam a visão existente sobre DX durante um cenário de mudanças, indicando os caminhos a serem seguidos para a comunidade científica e profissional.

## 1.1. Objetivos

### 1.1.1. Objetivo geral

Este estudo objetivou-se à realização de uma pesquisa de *Grey Literature Review* (GLR), conforme diretrizes presentes em Garousi *et al.* (2019). Visou-se identificar e caracterizar o conhecimento acerca de DX adquirido e divulgado por profissionais da indústria no contexto da pandemia do COVID-19 (2020-2022). O objetivo da pesquisa foi elaborado conforme modelo GQM (*Goal-Question-Metric*) apresentado na Tabela 1 [Basili, 1994].

**Tabela 1. Objetivo de Pesquisa - GQM**

<b>Analisar</b>	<i>developer experience</i>
<b>Com o propósito de</b>	caracterizar o objeto de pesquisa
<b>Com respeito a</b>	definições, métricas e ferramentas
<b>Sob o ponto de vista de</b>	profissionais da indústria
<b>No contexto de</b>	publicações em literatura cinza

### 1.1.2. Objetivos específicos

Os objetivos específicos deste trabalho envolvem a identificação e caracterização de definições, métricas e ferramentas no domínio de DX. Para tanto, elaboraram-se quatro questões de pesquisas (QPs), apresentadas abaixo:

- **QP1 - Qual a definição de DX adotada pela indústria?** Razão: Analisar a conceituação adotada pela indústria na propagação de conhecimentos sobre DX.
- **QP2 - Quais as ferramentas de apoio de DX?** Razão: Investigar o uso de ferramentas de apoio à promoção de DX e o cenário no qual são utilizadas.
- **QP3 - Quais métricas são utilizadas para DX? Como são coletadas?** Razão: Identificar métricas de DX e meios de coleta com base nas ferramentas identificadas.

## 2. Metodologia

Nylund (2020) e Greiler *et al.* (2023) sugerem a aplicação de estudos para investigação de DX sob a ótica de profissionais da indústria como complemento a sua pesquisa. A

partir desse pressuposto, a presente pesquisa busca evidenciar a caracterização de DX sob essa perspectiva. Para tanto, utilizou-se o método GLR (*Grey Literature Review*), conforme indicação presente no estudo de Garousi *et al.* (2019).

O estudo de Garousi *et al.* (2019) define três categorias de literatura com base no seu grau de credibilidade: branca, cinza e preta. A literatura branca (WL, *White Literature*) compreende publicações em conferências, livros, revistas e similares, portanto envolvem um alto grau de credibilidade. A literatura cinza (GL, *Grey Literature*), por sua vez, compreende relatórios técnicos, *pré-prints*, áudio e vídeo, palestras e postagens em *blogs*, com um grau menor de credibilidade. Já a literatura preta (BL, *Black Literature*), compreende ideias, conceitos e pensamentos, veiculados em diferentes formatos e que não possuem uma credibilidade nativamente identificável.

Williams e Rainer (2017) evidenciam que o conhecimento dos profissionais da indústria são oportunidades para a comunidade científica de explorar a experiência de mercado sobre determinados temas. Falam ainda que uma das formas de captar e analisar esse conhecimento é através das fontes de literatura cinza (GL, *Grey Literature*).

## 2.1. Base de Dados

Uma das fontes consideradas como pertencentes ao conjunto de GL são as postagens em *blogs* [Garousi *et al.*, 2019]. A partir das postagens, é possível aumentar a visibilidade que se tem sobre engenharia de *software*, por meio da visão dos autores [Williams e Rainer, 2017; Storey *et al.*, 2014]. *Blogs* também representam 28,3% das fontes utilizadas por pesquisadores brasileiros em engenharia de *software*, de acordo com o estudo de Kamei *et al.* [2022].

A fonte utilizada para a realização deste trabalho foi o *Medium*<sup>1</sup>. Uma plataforma aberta com mais de 100 milhões de leitores<sup>2</sup> para publicação de artigos no formato *blog*, identificada como fonte em outros trabalhos de revisões literárias [Brito, Terra e Valente, 2018; Williams, 2019; Oliveira *et al.*, 2021].

## 2.2. Palavra-chave

Esta revisão literária ocorreu entre Agosto de 2022 e Abril de 2023. A plataforma *Medium* não possui pesquisa avançada, contendo apenas uma entrada de texto para inserção do texto a ser pesquisado. Dada a limitação da ferramenta, o termo de pesquisa utilizado foi “*developer experience*”.

## 2.3. Critérios de Seleção

O termo de pesquisa retorna diversos resultados, estes foram filtrados através da aplicação de critérios de seleção:

- **CSI - Acesso gratuito e ilimitado.** O repositório utilizado para coleta de dados possui conteúdos de acesso gratuito ilimitado, e conteúdos de acesso por meio de assinatura. Por não estarem disponíveis para acesso gratuito e ilimitado, alguns dos textos não foram selecionados.

---

<sup>1</sup> <https://medium.com/>

<sup>2</sup> <https://medium.com/about> - acesso em jun. 2023

- **CS2 - Escrita em língua inglesa.** Foram analisados textos publicados em língua inglesa para analisar textos com um maior potencial de alcance a comunidade de engenharia de *software* a nível global.
- **CS3 - Conteúdo sobre DX.** Critério para exclusão de textos recuperados pela ferramenta que apresentam fuga ao tema.
- **CS4 - Credibilidade do(s) autor(es) atestada.** Uma das dificuldades em GLR é a credibilidade da fonte analisada [Kamei *et al.*, 2022]. Kamei *et al.* [2020] citam que a experiência pode ser um critério de credibilidade a ser utilizado ao realizar GLR. A credibilidade de ao menos um dos autores será atestada pela verificação de experiência em áreas correlatas à ciência da computação no seu perfil do LinkedIn<sup>3</sup>.
- **CS5 - Estar no intervalo de análise.** O intervalo de análise compreende publicações realizadas entre 2020 e 2022. O período foi selecionado pelos impactos que a pandemia causou no aumento da população de desenvolvedores e nas relações de desenvolvedores [GitHub, 2020; Neto *et al.*, 2021].

## 2.4. Revisão de Literatura

A execução desta GLR foi dividida em etapas: (1) Extração; (2) Seleção; e, (3) Análise.

Para a etapa de *Extração*, utilizou-se um script de *web scrapping* desenvolvido em *NodeJS*. Como saída, os dados foram enumerados e estruturados em um arquivo JSON contendo informações como título, autor, idioma, data de publicação e *link*. Para classificação do idioma do texto, o script consumiu o serviço de inteligência artificial da *Microsoft Azure* para análise de texto, *Cognitive Services/Text Analytics*<sup>4</sup>.

Para a etapa de *Seleção*, o *CSI (texto gratuito)* foi aplicado mediante verificação unitária dos textos recuperados. O *CS2 (texto em inglês)* foi aplicado através da seleção realizada pelo serviço de análise de texto da *Microsoft*. Para os textos em que o idioma não foi reconhecido foram analisados individualmente pelo autor. O *CS5 (intervalo de publicação)* foi aplicado pela filtragem da data de publicação extraída pelo script. Já o *CS4 (credibilidade)* foi aplicado por meio da identificação dos perfis dos autores no corpo do texto, em seu perfil no *Medium* e pela pesquisa de seus perfis no *LinkedIn*.

Para a aplicação do *CS3* (fuga ao tema) este trabalho contou com a revisão em pares de colaboradores, dois alunos de mestrado do Programa de Pós-graduação em Ciência da Computação da Faculdade de Computação da UFMS (PPGCC/FACOM) que pesquisam sobre DevRel.

Para a etapa de *Análise*, utilizou-se uma adaptação da abordagem descrita por Cruzes e Dyba [2011]:

1. Leitura e releitura dos textos: leitura de todos os textos pelo autor e de parte do *dataset* pelos colaboradores (distribuídos aleatoriamente);
2. Extração das respostas para as questões de pesquisa: pelo autor e pelos colaboradores, conforme divisão do *dataset*;
3. Combinação dos resultados: combinação dos resultados da etapa 2 para cada questão de pesquisa;

<sup>3</sup> <https://www.linkedin.com/>

<sup>4</sup> <https://azure.microsoft.com/pt-br/products/cognitive-services/text-analytics>

4. Análise de resultados: análise quantitativa e qualitativa dos resultados com suporte no referencial teórico utilizado.

Durante todas as etapas da pesquisa, houve supervisão do orientador sobre os métodos empregados e discussões entre os responsáveis pela análise para alinhamentos sobre o processo de síntese utilizado.

### 3. Resultados

Através da *string* de pesquisa ‘*developer experience*’, obteve-se um retorno de 889 registros de publicações. Destes, 514 registros compreendem o recorte temporal analisado (CS5), dos quais 135 foram publicados em 2020, 167 em 2021 e 212 em 2022 (Figura 4). A Figura 5 exibe um fluxograma do total de textos obtidos em cada etapa de aplicação dos critérios de seleção.

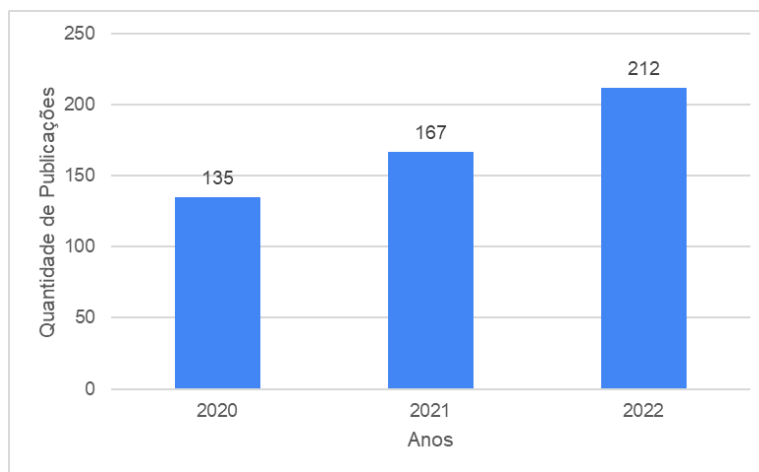


Figura 4. Distribuição de Registros por Ano

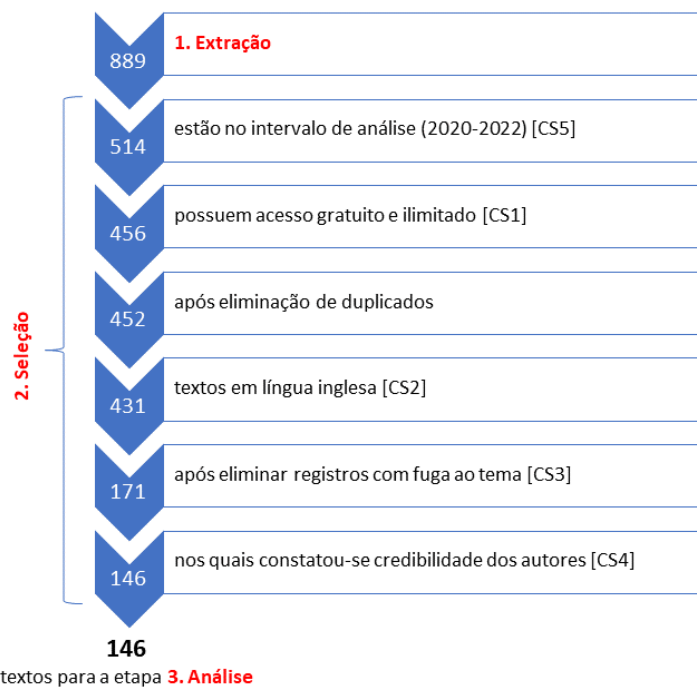


Figura 5. Fluxograma de Aplicação dos Critérios de Seleção

Dos textos presentes no intervalo analisado neste trabalho, 456 foram utilizados para aplicação dos critérios de seleção. Uma vez que 58 possuem acesso pago e foram eliminados pela aplicação do *CSI*. A partir desta seleção, obtivemos 452 textos nos quais os demais critérios de seleção foram aplicados. Quatro registros do número inicial obtido foram excluídos por serem publicações duplicadas.

Aplicando o critério de seleção *CS2*, obtivemos 431 textos publicados em língua inglesa. Por meio do *CS3*, 260 registros foram removidos por fuga ao tema, uma vez que o buscador do repositório utilizado retorna todos os registros que contenham: (a) marcadores com o termo; (b) a correspondência exata ou não da *string* de pesquisa, retornando deste modo textos que contenham as duas palavras, seja em seu título, corpo do texto ou nome do autor; (c) além de retornar também comentários realizados em publicações.

Por meio do critério *CS4*, a credibilidade dos autores foi atestada através da identificação de trabalho ou formação em áreas correlatas à ciência da computação, disponíveis em seus perfis do *LinkedIn*. Dos 171 registros, até então, foi possível identificar e atestar a credibilidade de um ou mais autores de 146 textos.

Nas próximas seções, são descritas as respostas obtidas para cada questão de pesquisa apresentada como objetivos específicos deste trabalho. Para indicar a fonte dos resultados, utilizamos indicadores dos textos revisados da base de dados<sup>5</sup>.

### 3.1. QP1 - Qual a definição de DX adotada pela indústria?

Dos 146 registros que foram para a etapa de extração de dados, 31 continham definições para DX (Tabela 2). Compreendeu-se todos os textos nos quais o(s) autor(es) indicavam que definiam ou compreendiam DX, além de textos que buscavam significar o termo ‘*Developer Experience*’. (ID29, ID30, ID42, ID49, ID51, ID58, ID65, ID66, ID77, ID78, ID84, ID272, ID275, ID277, ID284, ID306, ID307, ID329, ID348, ID350, ID355, ID368, ID370, ID397, ID415, ID474, ID548, ID634, ID733, ID802, ID838).

**Tabela 2. Definições de DX**

Definição	Ocorrências
Comparada a UX	ID29, ID65, ID77, ID78, ID84, ID284, ID306, ID307, ID348, ID350, ID355, ID368, ID415, ID802
Cultura, processos, ferramentas e arquitetura	ID397
Felicidade	ID78, ID277, ID329, ID397, ID548, ID733
Redução no tempo de execução de tarefas específicas ou como promotor de agilidade	ID42, ID66, ID84, ID277, ID284, ID329
Separação entre aspectos internos e externos	ID58, ID77, ID370
Comparada a CX	ID838

<sup>5</sup> <https://github.com/gabriel-rabelo/tcc-gabriel>



Destes, 14 textos confrontam DX (*Developer Experience*) com UX (*User Experience*) (ID29, ID65, ID77, ID78, ID84, ID284, ID306, ID307, ID348, ID350, ID355, ID368, ID415, ID802). Os autores se dividem entre: (a) a defesa de que DX compreende a experiência do usuário de um produto de *software*, enquanto esse usuário é um desenvolvedor; e (b) a defesa de que seria a experiência de um usuário não ao consumir, mas sim em desenvolver um produto. O ponto central da discussão é o entendimento de que DX engloba não somente aspectos técnicos ou tecnológicos. Ela engloba também aspectos culturais e organizacionais, dado que o foco de DX é fornecer bons produtos tanto quanto boas condições para entregas de qualidade.

Uma definição interessante segrega DX em quatro domínios: *cultura*, *processos*, *ferramentas* e *arquitetura* (ID397). Nota-se que, *cultura*, para o autor, é definida enquanto agente para promoção de segurança para o time de desenvolvimento, como forma de seguridade de felicidade geral. O termo felicidade é observado em outras ocorrências, assim como aspectos emocionais. Eles conversam diretamente com o aspecto de harmonia organizacional, essencial para a manutenção da cultura. Isso é amplamente observado entre as *BigTechs*<sup>6</sup>, que aumentaram o foco em saúde mental<sup>7</sup>, termo também em voga nos dias correntes (ID78, ID277, ID329, ID397, ID548, ID733).

Há ocorrências indicando DX como um agente responsável pela redução no tempo de execução de tarefas específicas ou como promotor de agilidade (ID42, ID66, ID84, ID277, ID284, ID329). A redução ocorre no contexto do processo de entrega (início ao fim do ciclo de desenvolvimento), garantindo assim, a eficiência das equipes. Existem menções sobre a integração entre ciclos de desenvolvimento *interno* e *externo* (ID58, ID77, ID370). Interno, no qual o desenvolvedor executa uma tarefa para um entregável. E *externo*, na integração entre os entregáveis de todos os membros do time de desenvolvimento da organização.

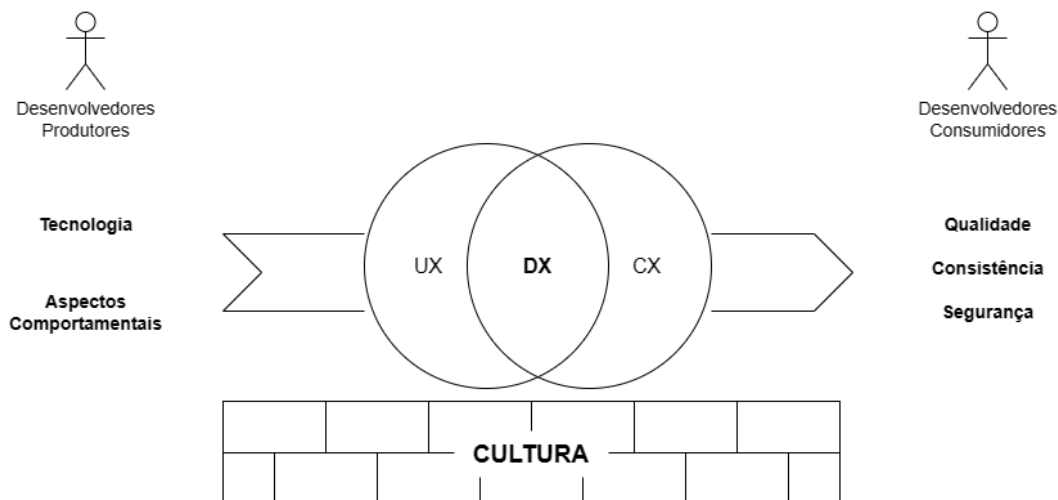
Existe também uma ocorrência correlacionando DX com CX (*Customer Experience*) (ID838). A correlação é com foco na construção de relacionamentos visando uma interação a longo prazo, aplicada a DX externa. Os demais autores abordam o aspecto global em desenvolvimento ou em consumo de *software*, de modo mais generalista que os demais (ID30, ID49, ID51, ID272, ID275, ID474, ID634).

Sucintamente, a partir da visão de profissionais da indústria, DX é o ponto integrador entre os pilares *cultura*, *processos* e *ferramentas*. Com foco no fornecimento de uma *experiência* satisfatória àqueles que participam da construção de um *software* (análogo a UX) ou que utilizam-no à elaboração de outros, como clientes (análogo a CX). Assim sendo, DX é a intersecção entre UX e CX como parte central da cadeia de desenvolvimento (Figura 6). É alimentada por facilitadores *tecnológicos* e *comportamentais*, sustentados por uma base *cultural*. Tem como objetivo fornecer *qualidade*, *consistência* e *segurança* aos consumidores finais.

---

<sup>6</sup> [https://en.wikipedia.org/wiki/Big\\_Tech](https://en.wikipedia.org/wiki/Big_Tech)

<sup>7</sup> <https://exame.com/bussola/rh-de-2023-cuida-de-saude-mental-e-qualidade-de-vida/>



**Figura 6. Cadeia de Desenvolvimento no aspecto de Developer Experience**

A proposta de desenho para a Cadeia de Desenvolvimento, com ênfase em DX é complementar às definições de Fagerholm (2015) e Greiler *et al.* (2023). Fagerholm (2015) define DX como a tríade *conhecimento, afeto e conação*. Descreve sua inserção na ponte existente entre o aspecto técnico e humano do processo de desenvolvimento de *software*. Já Greiler *et al.* (2023), exploram os aspectos humanos na definição de DX sobre a segmentação de *expectativas, percepções e sentimentos*. Estes são definidos como fatores desencadeadores lotados em aspectos técnicos e comportamentais. Há destaque a cultura, como o principal fator de importância para os desenvolvedores.

DX pode ser vista como a união entre UX e CX, os três conceitos se generalizam como a experiência humana frente a um objeto, no caso proposto, um *software*. A experiência, por si só, envolve o desencadear de ações realizadas por um indivíduo. Estas com base em conhecimentos inatos a cada um, e os aspectos afetivos envolvidos nessas ações, sejam eles expectativas, percepções ou sentimentos. Essa caracterização envolve algo intrínseco ao ser humano. No entanto, toda e qualquer experiência é afetada por fatores externos, ou seja, o ambiente em que se insere. Por ser algo intangível, não é fácil definir uma experiência. Além disso, é algo que pode ser observado por diferentes pontos de vista, cada qual com um foco sobre o objeto estudado. Nota-se que, é possível confrontar as definições de modo complementar em um único diagrama (Figura 7). Este é um conceito de DX que une os diferentes ângulos pelo qual DX é observada e analisada com a visão que observamos.



ID415, ID424, ID427, ID461, ID462, ID489, ID537, ID590, ID622, ID743, ID786, ID818, ID829, ID838, ID839, ID846, ID861, ID874, ID875, ID900, ID914).

**Tabela 3. Ferramentas para análise de DX**

<b>Ferramenta</b>	<b>Ocorrências</b>
DX Scanner	ID900
Observação da Comunidade ou Ecossistema	ID10, ID19, ID28, ID29, ID46, ID49, ID50, ID66, ID77, ID78, ID91, ID97, ID98, ID139, ID198, ID201, ID207, ID208, ID211, ID213, ID214, ID241, ID261, ID277, ID291, ID306, ID314, ID352, ID353, ID366, ID370, ID396, ID402, ID415, ID424, ID427, ID461, ID462, ID489, ID537, ID590, ID622, ID743, ID818, ID829, ID838, ID846, ID861, ID875, ID914
Formação e acompanhamento da comunidade	ID19, ID28, ID46, ID49, ID78, ID98, ID139, ID198, ID201, ID261, ID291, ID306, ID314, ID353, ID366, ID415, ID461, ID489, ID537, ID590, ID622, ID743, ID838, ID846, ID914
GitHub Issues	ID28, ID139, ID201, ID461
Twitter	ID28
Reddit	ID198
User Research	ID10, ID66, ID139, ID201, ID207, ID208, ID211, ID213, ID214, ID277, ID366, ID402, ID427, ID462, ID537, ID818, ID829, ID846, ID861, ID875
Personas	ID50, ID77, ID427
Mapeamento de jornadas (usuários e desenvolvedores)	ID29, ID46, ID91, ID241, ID396, ID424, ID427
Ciclos de Feedback	ID84, ID91, ID139, ID277, ID370, ID415, ID489, ID537, ID818, ID846, ID914
Canais de comunicação	ID28, ID29, ID78, ID139, ID198, ID201, ID914
Service Desk	ID29
Ferramentas de logging e analytics	ID29, ID68, ID97, ID489, ID786, ID839, ID874
Hotjar	ID29, ID97
Mixpanel	ID29, ID97
Grafana	ID839, ID874

Dentre todas as 56 pesquisas, uma menciona uma ferramenta automatizada à análise de DX sobre um produto, o *DX Scanner*<sup>8</sup> (ID900). Ao conferir o landing *page* da ferramenta, é possível observar que ela apresenta indicadores diversos sobre o produto. Existe ainda a possibilidade de personalização das funcionalidades a partir da contratação de um plano pago. As demais ferramentas mencionadas nos textos são utilizadas como apoio para a análise.

Um ecossistema é formado pela interação entre uma comunidade (pessoas) e produtos. Formar e acompanhar uma comunidade possibilita identificar tendências e compreender dores e alívios em relação a um produto. Ocorrem menções em 50 publicações de interações diretas ou indiretas, por meio de observação, com a comunidade ou ecossistema (ID10, ID19, ID28, ID29, ID46, ID49, ID50, ID66, ID77, ID78, ID91, ID97, ID98, ID139, ID198, ID201, ID207, ID208, ID211, ID213, ID214, ID241, ID261, ID277, ID291, ID306, ID314, ID352, ID353, ID366, ID370, ID396, ID402, ID415, ID424, ID427, ID461, ID462, ID489, ID537, ID590, ID622, ID743, ID818, ID829, ID838, ID846, ID861, ID875, ID914).

Das 50 publicações sobre comunidade ou ecossistema, 25 citam a formação e o acompanhamento de comunidade (ID19, ID28, ID46, ID49, ID78, ID98, ID139, ID198, ID201, ID261, ID291, ID306, ID314, ID353, ID366, ID415, ID461, ID489, ID537, ID590, ID622, ID743, ID838, ID846, ID914). Do conjunto, cinco itens citam o acompanhamento da comunidade em fóruns de discussão e redes sociais (ID28, ID139, ID198, ID201, ID461). *GitHub Issues*<sup>9</sup> pode ser encontrado em quatro destes textos (ID28, ID139, ID201, ID461), enquanto *Twitter* (ID28) e *Reddit* (ID198) são encontrados em apenas um texto, cada.

Em relação a realização de *User Research*, através de questionários ou entrevistas, existem 29 citações dentre as 50 (ID10, ID66, ID139, ID201, ID207, ID208, ID211, ID213, ID214, ID277, ID366, ID402, ID427, ID462, ID537, ID818, ID829, ID846, ID861, ID875). Em relação a criação de personas, existem três (ID50, ID77, ID427). Por sua vez, o mapeamento de jornada, seja de usuário ou desenvolvedor, é observado em sete (ID29, ID46, ID91, ID241, ID396, ID424, ID427). Internamente, nas organizações, a observação pode ser realizada por meio de ciclos de *feedback*. Possibilitando avaliar diferentes aspectos em diferentes domínios, sejam eles processos ou ferramentas. Rotinas recorrentes de *feedback* estão presentes em 11 publicações (ID84, ID91, ID139, ID277, ID370, ID415, ID489, ID537, ID818, ID846, ID914).

A observação e o contato com a comunidade é realizada por meio de canais de comunicação existentes na comunidade de usuários, observados em sete publicações (ID28, ID29, ID78, ID139, ID198, ID201, ID914). A formação de um canal de contato para informar um *bug* sobre o sistema, como um *service desk*, é mencionado em uma destas publicações (ID29).

Outra estratégia de observação é o uso de ferramentas de estatísticas, monitoramento e rastreamento para coleta de dados, mais conhecidas como ferramentas de *logging* e *analytics*, que estão presentes em sete publicações (ID29, ID68, ID97, ID489, ID786, ID839, ID874). *Hotjar* e *Mixpanel* são observadas em duas publicações (ID29, ID97), assim como o *Grafana* (ID839, ID874).

---

<sup>8</sup> <https://dxscanner.io/>

<sup>9</sup> <https://github.com/features/issues>

A partir do contato, *feedback* e observação da comunidade, dados são coletados e por consequência formam um importante ativo para o negócio. A posse desses dados é valiosa e pode ser utilizada para extrair informações para a formação de um conhecimento. Esse conhecimento pode ser utilizado nas estratégias de DX para melhorias no posicionamento do produto no mercado e para melhorias nos processos internos de desenvolvimento de *software*.

### 3.2.2. Ferramentas para promoção DX

Dos 136 textos que abordam ferramentas, 112 abordam ferramentas para melhoria de DX em relação a um produto (Tabela 4) (ID2, ID11, ID18, ID19, ID35, ID38, ID42, ID49, ID50, ID51, ID53, ID57, ID58, ID65, ID68, ID74, ID77, ID78, ID82, ID84, ID91, ID94, ID97, ID98, ID108, ID118, ID120, ID124, ID159, ID163, ID167, ID172, ID181, ID206, ID207, ID208, ID209, ID211, ID219, ID222, ID225, ID237, ID243, ID252, ID254, ID257, ID258, ID269, ID273, ID275, ID278, ID279, ID283, ID295, ID299, ID307, ID314, ID329, ID348, ID350, ID352, ID353, ID361, ID366, ID370, ID383, ID394, ID395, ID396, ID402, ID410, ID415, ID418, ID424, ID425, ID473, ID485, ID489, ID499, ID506, ID548, ID559, ID576, ID590, ID594, ID622, ID634, ID652, ID713, ID733, ID743, ID758, ID784, ID786, ID792, ID802, ID818, ID829, ID830, ID838, ID839, ID841, ID846, ID853, ID859, ID860, ID861, ID864, ID874, ID912, ID919, ID949).

**Tabela 4. Ferramentas para promoção de DX**

Ferramenta	Ocorrências
Documentação	ID18, ID42, ID51, ID53, ID65, ID78, ID98, ID225, ID257, ID273, ID275, ID278, ID279, ID307, ID314, ID329, ID350, ID353, ID370, ID394, ID395, ID425, ID489, ID559, ID590, ID594, ID634, ID733, ID743, ID786, ID792, ID802, ID838, ID841, ID859, ID860, ID861, ID919
OpenAPI, Swagger e Redocly	ID78, ID594
Code Samples e Code Snippets	ID53, ID78, ID208, ID219, ID254, ID257, ID307, ID396, ID415, ID424, ID733, ID802, ID829, ID838
Tutoriais	ID257, ID383, ID802
Portal do desenvolvedor	ID11, ID38, ID50, ID53, ID91, ID237, ID252, ID257, ID269, ID396, ID485, ID634, ID652, ID818, ID912, ID919, ID949
Backstage.io	ID269, ID949
Autosserviço	ID78
Free Trial	ID77, ID98, ID353, ID622

<b>Ferramenta</b>	<b>Ocorrências</b>
Sandbox	ID314, ID395, ID838
Facilidade de uso e Onboarding	ID78, ID97, ID167, ID424, ID792, ID818
SDKs, frameworks, bibliotecas, plugins e pacotes	ID19, ID35, ID78, ID98, ID108, ID163, ID206, ID207, ID211, ID225, ID279, ID283, ID307, ID314, ID329, ID394, ID634, ID652, ID802, ID919
Bibliotecas e componentes reutilizáveis	ID35, ID211, ID295, ID548, ID864
Presença multiplataforma	ID78, ID159, ID211
IDE	ID57, ID74, ID108, ID159, ID181, ID211, ID222, ID329, ID758, ID786, ID841, ID853
Hot reload	ID108, ID211
Autocomplete e highlighting	ID181, ID211, ID841
Depuradores	ID108, ID222, ID841, ID853
Customer Experience	ID254, ID258
Workshops e conferências	ID299, ID366, ID402, ID846
Consultoria	ID366
Auditoria	ID49
Ciclo PDCA e 5Ws	ID396, ID839
Design Thinking	ID402
Design-First	ID830
Figma	ID348
AdobeXD	ID348
Sketch	ID348
Grammer	ID348
Gifox	ID348
Icordingon	ID348
Infraestrutura de TI	ID42, ID329, ID792
Orquestradores de contêineres	ID84, ID499

<b>Ferramenta</b>	<b>Ocorrências</b>
Kubernetes	ID499
Simuladores de ambiente cloud	ID410, ID713
Skaffold	ID410
DevSpace	ID410
Tilt	ID410
Heroku	ID243
Vercel	ID283
Arquitetura de Software	ID58, ID82, ID209, ID299, ID353, ID425, ID548, ID864, ID949
Microserviços	ID58, ID299, ID353, ID949
OpsLevel	ID949
Yoshi	ID548, ID864
Automação de tarefas	ID42, ID51, ID65, ID118, ID172, ID273, ID350, ID361, ID370, ID410, ID418, ID473, ID576, ID784, ID839, ID874, ID949
Pipeline CI/CD	ID118, ID273, ID361, ID576, ID784, ID839, ID874, ID949
Gitlab CI	ID874
Jenkins CI	ID949
Circle CI	ID949
Travis CI	ID949
GitHub Actions	ID576, ID949
Adoção de padrões	ID65, ID78, ID120, ID124, ID295, ID425, ID548, ID839
Controle de versionamento	ID124, ID295, ID839
Versionamento semântico	ID124, ID839
Design System ou Style Guide	ID120, ID425, ID548
RFCs	ID78



Ferramenta	Ocorrências
Conventional Commits	ID839
GitHub Flow	ID839
Continuous Deployments	ID839
Estratégias de testes	ID295
Gerenciamento de mudanças	ID124

Dentre a seleção de publicações que abordam ferramentas para melhoria de DX, menção a documentação, como uma ferramenta de apoio à DX, existe em 38 registros (ID18, ID42, ID51, ID53, ID65, ID78, ID98, ID225, ID257, ID273, ID275, ID278, ID279, ID307, ID314, ID329, ID350, ID353, ID370, ID394, ID395, ID425, ID489, ID559, ID590, ID594, ID634, ID733, ID743, ID786, ID792, ID802, ID838, ID841, ID859, ID860, ID861, ID919). Ferramentas e padrões para documentação como *OpenAPI*, *Swagger* e *Redocly*, são observadas em dois destes 38 textos (ID78, ID594).

Ainda no aspecto de documentação, *code samples* e *code snippets* são citados em 14 publicações (ID53, ID78, ID208, ID219, ID254, ID257, ID307, ID396, ID415, ID424, ID733, ID802, ID829, ID838). *Code samples* são exemplos de códigos para apoio à aprendizagem [Menezes *et al.*, 2022]. Já *Code Snippets*, são trechos de códigos, que são cópias de outros trechos disponibilizados na internet ou de sistemas [Ragkhitwetsagul *et al.*, 2019]. Os tutoriais, por sua vez, estão presentes em três textos (ID257, ID383, ID802). Documentação é um fator de suma importância para DX. Por meio dela, o primeiro contato do desenvolvedor com o produto ocorre, seja um produto interno ou externo, ou ainda um produto no qual o desenvolvedor contribui. Além do primeiro contato, a documentação serve como guia para consultas durante todas as etapas do relacionamento desenvolvedor-*software*.

Espera-se que a documentação de um *software* esteja localizada contígua às demais necessidades para o consumo do produto, como geração de chaves, *tokens*, e demais configuráveis. A localização, usualmente, encontra-se em portais para desenvolvedores, sejam eles internos ou externos, que são apontados em 17 registros (ID11, ID38, ID50, ID53, ID91, ID237, ID252, ID257, ID269, ID396, ID485, ID634, ID652, ID818, ID912, ID919, ID949). Dois destes registros relatam uma ferramenta que pode ser utilizada para a geração desses portais, o *Backstage.io* (ID269, ID949).

Apenas uma ocorrência aborda o conceito autosserviço (do inglês, *self-service*) para produtos (ID78). Por outro lado, quatro textos abordam o conceito *free trial*, no qual os usuários podem utilizar parte ou todo de um sistema, como oportunidade para testar o produto (ID77, ID98, ID353, ID622). Esses conceitos são correlacionados com outro, *sandbox*, presente em três citações (ID314, ID395, ID838). Para *software*, *sandbox* representa um espaço no qual desenvolvedores podem explorar e desbravar o potencial de uma aplicação em um ambiente controlado e isolado. Permitindo o preparo de provas de conceito, antes de aplicar a ferramenta em questão ao seu produto ou serviço [Jamalpur *et al.*, 2018].

O relacionamento entre usuário e produto é complexo e sugere a necessidade de estabelecer vínculos entre a expertise individual e as funcionalidades fornecidas. Dos modelos e práticas de *marketing*, um dos mais conhecidos é o Modelo AIDA<sup>10</sup>, elaborado na virada para o século XX [Londero, 2017]. O modelo descreve que uma adesão a um produto procede da conquista de *atenção*, a provocação da *intenção*, o despertar de um *desejo* e a realização de uma *ação*, a adesão propriamente colocada. Ao prover um espaço seguro para testes, sob a ótica externa de DX, abre-se espaço para práticas que levam à conversão. Por ser justamente um espaço onde a exploração do potencial do produto gera um desejo de uso, conseqüentemente, esse desejo poderá ser concretizado por meio do consumo de uma ferramenta de *software*.

Uma prática que facilita essa conversão, ao tratarmos ainda de DX externa, é o *onboarding* de produtos, o processo do primeiro contato ao primeiro uso. *Onboarding* também é observado no aspecto interno de DX, desde a chegada de um colaborador à uma equipe até sua primeira contribuição. Esses processos devem ser claros e em um nível de dificuldade que leve a promover um crescimento acelerado da curva de aprendizagem sobre a ferramenta. A facilidade de uso ou *onboarding* ocorre em seis textos que mencionam ferramentas (ID78, ID97, ID167, ID424, ID792, ID818).

De outro lado, algo que prejudica a concretização da ação, na qual se opta pelo uso de uma ferramenta, é sua compatibilidade, ou falta dela, com as demais do mercado. Atualmente, novos produtos de *software* nascem todos os dias, assim como chegam ao fim de sua vida. Um dos fatores que determinam a perenidade de um produto é como esse se relaciona com os outros, que já dominam uma parcela do mercado. SDKs (*Software Development Kits*, Kit de Desenvolvimento de Software), *frameworks*, bibliotecas, *plugins* e pacotes são essenciais para a presença de mercado. Essas ferramentas permitem a integração com fornecedores terceiros, ou seja, sua compatibilidade, fator citado em 20 casos (ID19, ID35, ID78, ID98, ID108, ID163, ID206, ID207, ID211, ID225, ID279, ID283, ID307, ID314, ID329, ID394, ID634, ID652, ID802, ID919).

Internamente, a criação de bibliotecas e componentes reutilizáveis reduz práticas de re-trabalho e aumentam a colaboração entre diferentes atores dentro do processo de desenvolvimento. Cinco publicações retratam essa prática como um agente para promoção de DX (ID35, ID211, ID295, ID548, ID864). Outro aspecto em relação a presença de mercado e na comunidade é a compatibilidade com diferentes plataformas. Três ocorrências indicam a presença multiplataforma como uma ferramenta potencializadora de DX (ID78, ID159, ID211).

IDEs (*Integrated Development Environments*, Ambientes Integrados de Desenvolvimento) são observadas como ferramentas de DX em 12 textos, conjuntamente à ferramentas, compatibilidades e funcionalidades presentes ou integradas a ela (ID57, ID74, ID108, ID159, ID181, ID211, ID222, ID329, ID758, ID786, ID841, ID853). Há duas ocorrências sobre IDE baseada em navegadores (ID57, ID758), sendo uma delas sobre o *GitHub Codespaces*<sup>11</sup> (ID758). Outras duas mencionam *hot reload* (ID108, ID211). *Autocomplete* e *highlighting* são observadas em três ocorrências (ID181, ID211, ID841). Já as ferramentas para depuração são observadas em quatro (ID108, ID222, ID841, ID853).

---

<sup>10</sup> [https://en.wikipedia.org/wiki/AIDA\\_\(marketing\)](https://en.wikipedia.org/wiki/AIDA_(marketing))

<sup>11</sup> <https://github.com/features/codespaces>

Essas ferramentas integram o processo primário do principal entregável de produtos de *software*, o código-fonte. Existem no mercado diversas opções de IDEs, com diversos componentes que auxiliam o desenvolvedor como facilitadores na atividade de programação. *Customer Experience* é a experiência de consumo de um produto, é essa experiência que leva um consumidor a optar por um produto ou por um concorrente. Existem duas publicações que retratam a experiência da relação de consumo entre desenvolvedores e *software* (ID254, ID258).

Em relação a experiência de depuração de código, encontrar a causa raiz de um *bug* é uma atividade que exige uma grande capacidade analítica proporcional a quantidade de linhas de código existentes. Uma publicação menciona o uso de aprendizagem de máquina para processamento de código-fonte para a identificação de *bugs* em grandes aplicações (ID2). O processamento de dados, como a análise via aprendizagem de máquina e outros métodos, está presente na realidade de muitas organizações. Com o advento da internet aumentou-se a quantidade de dados gerados e disponibilizados na rede, estes dados podem ser utilizados para a tomada de decisão. Decisões guiadas a partir de dados estão presentes em duas publicações (ID2, ID415).

Além das questões de consumo, externas, ações internas visando DX para a produção de *software* podem ser realizadas, como *workshops* e conferências. Estas ações são mencionadas em quatro textos (ID299, ID366, ID402, ID846). Além disso, um dos textos apresenta a possibilidade de contratação de uma equipe de consultoria externa para promoção de ações de DX (ID366). Outra ação interna é a de auditoria, presente em uma ocorrência (ID49). Ela pode ser utilizada para verificar o cumprimento das normativas, regras e princípios de DX pela organização (ID49).

Dois publicações retratam o uso de ferramentas da área de negócios e administração para elaboração de estratégias de DX, Ciclo PDCA e 5Ws (ID396, ID839). Por sua vez, da área de *design*, uma publicação cita o método *Design Thinking* (ID402) e uma cita a abordagem *Design-First* (ID830). Além de métodos, outra publicação cita ferramentas de prototipação e *design* (ID348), são elas: *Figma*, *AdobeXD*, *Sketch*, *Gramer*, *Gifox*, *Icordingon*.

Ações como consultorias ou abordagens de design necessitam de negociação como uma habilidade. Uma publicação vislumbra a habilidade de negociação como uma ferramenta para DX (ID352). Negociação vai além de DX, torna-se um meio necessário para o convívio social e para o aspecto do trabalho, de modo geral; para um bom convívio é indispensável a harmonia. Visando a harmonia e o bem-estar, sob a ótica do trabalho presencial, uma publicação menciona a criação de aconchego através de músicas ambiente (ID94).

Para além do bem-estar, as organizações devem fornecer aos seus trabalhadores ferramentas de trabalho, sejam elas computacionais ou não, de acordo com suas políticas. Três publicações citam a infraestrutura organizacional (ID42, ID329, ID792). Uma delas cita a máquina individual de trabalho já configurada, sem necessidade de atuação do desenvolvedor na instalação ou configuração de itens básicos como VPN (*Virtual Private Network*, Rede Privada Virtual), SDK, IDE e outros (ID792). Já a outra cita poder de processamento (ID329). Além da máquina, acessórios de escritório e periféricos são citados (ID329).

No que tange infraestrutura de servidores, cinco publicações mencionam computação em nuvem como facilitadora (ID57, ID68, ID243, ID283, ID506). Duas

publicações abordam orquestradores de contêineres como ferramenta para gerenciamento (ID84, ID499). *Kubernetes* é explicitamente mencionado em apenas uma (ID499). A simulação em ambiente local de arquitetura em nuvem é mencionada em dois textos como um desafio a ser enfrentado para realização de testes (ID410, ID713). As ferramentas *Skaffold*, *DevSpace* e *Tilt* são mencionadas em um deles como uma possível solução (ID410). Duas plataformas de hospedagem em nuvem são mencionadas em diferentes publicações, são elas *Heroku* (ID243) e *Vercel* (ID283), atuando como facilitadores no processo de publicação e hospedagem. Os autores das publicações mencionam essas ferramentas como facilitadores das rotinas de DevOps. A cultura desse movimento ganha forças conjuntamente com o movimento ágil e em face dessa propagação surgem desafios. Muitas organizações associaram as práticas DevOps a implantação de computação em nuvem e, conseqüentemente, suas ferramentas [Leite *et al.*, 2019; Khan *et al.*, 2022; Keenan, Kersten e O’Connel, 2023]

De encontro com a computação em nuvem, temos questões arquiteturais, mencionadas em nove textos (ID58, ID82, ID209, ID299, ID353, ID425, ID548, ID864, ID949). Arquitetura desacoplada, ou microsserviços, são mencionadas em quatro destes (ID58, ID299, ID353, ID949). Destes, um menciona um gerenciador de microsserviços, o *OpsLevel* (ID949). Dois textos mencionam arquiteturas de *microfrontend* (ID209, ID548). Sendo que apenas um deles menciona uma ferramenta que pode ser utilizada como orquestrador para a criação de *microfrontend*, o *Yoshi*. Reuso e compartilhamento de recursos, como pela criação de componentes, são citados em dois textos (ID548, ID864). Apenas uma citação refere-se às práticas de Código Limpo (ID425). Há, ainda, uma citação sobre uma arquitetura para *blockchain*, a arquitetura XCVM (ID82).

A atividade de desenvolver software não é trivial. Envolve complexidades lógicas e matemáticas em diversas nuances, além da execução de inúmeras tarefas, como testes, compilação, implantação e outras. Existem 17 publicações que referem-se a automação de tarefas (ID42, ID51, ID65, ID118, ID172, ID273, ID350, ID361, ID370, ID410, ID418, ID473, ID576, ID784, ID839, ID874, ID949). Destas, oito mencionam automação para *pipelines* de CI/CD (*Continuous Integration/Continuous Delivery*, Integração Contínua/Entrega Contínua) (ID118, ID273, ID361, ID576, ID784, ID839, ID874, ID949). Uma publicação cita a ferramenta *Gitlab CI* (ID874) e outra cita as ferramentas *Jenkins CI*, *Circle CI*, *Travis CI* (ID949). O *GitHub Actions*, não exclusivo para CI, é citado em duas destas publicações (ID576, ID949). Destas, ainda existem três que mencionam automação de testes (ID51, ID172, ID273) e uma que menciona automação em escalonamento de recursos de infraestrutura (ID410). Cabe ressaltar que, um dos textos aborda a promoção da cultura *DevOps* dentro dos times de desenvolvimento, sob o conceito *self-service* (ID410).

Visando facilitar o cotidiano, a adoção de padrões auxilia na promoção de uma boa experiência e é observada em nove ocorrências (ID65, ID78, ID120, ID124, ID295, ID425, ID548, ID839). Destas, a padronização de controle de versionamento é observada em três (ID124, ID295, ID839), sendo o versionamento semântico abordado em duas (ID124, ID839). *Design System* ou *Style Guide* são observados também em três (ID120, ID425, ID548). Apenas uma ocorrência cita padronização de mensagens de erro por meio da adoção de RFCs (ID78). *Conventional Commits*, *GitHub Flow* e *Continuous Deployments* são citados apenas em uma ocorrência (ID839). De encontro com estes meios facilitadores, a adoção de estratégias para testes, não exclusiva para testes automatizados, é observada em uma ocorrência (ID295).

O controle de versionamento, abordado no último parágrafo, está envolvido em outro processo importante, o gerenciamento de mudanças, citado em apenas uma publicação (ID124). O texto aborda a estratégia de Expandir e Contratar para alterações de versões que promovem mudanças drásticas de comportamento de uma aplicação, como quando uma funcionalidade é depreciada. Alterações grandes, quando não passam por um bom gerenciamento, elevam o risco de evasão de usuários.

### 3.2.3. Ferramentas e DX

As ferramentas para DX, observadas após a análise dos textos selecionados, não são necessariamente ferramentas novas, como o *DX Scanner*. E sim ferramentas já existentes no processo de trabalho. Apesar das múltiplas ferramentas e áreas de aplicabilidade, a documentação segue como a principal e mais importante ferramenta. Em tempos de agilidade, ter um repositório único contendo um guia sobre o uso das principais funcionalidades e erros a serem encontrados poupa tempo do desenvolvedor. Além de estabelecer uma maior relação do usuário com a marca e permite acreditar uma maior confiabilidade sobre as soluções propostas.

De encontro com a questão de tempo, automação de tarefas se faz imprescindível. *Software* pode, além de poupar tempo de outros processos e profissões, poupar tempo de desenvolvimento e de desenvolvedores. Automação é uma temática crescente na atualidade e ainda há potencial a ser explorado, principalmente para fornecer uma boa experiência ao desenvolver ou consumir software.

### 3.3. QP3 - Quais métricas são utilizadas para DX? Como e por quem são coletadas?

Dentre os 146 textos selecionados, 39 mencionam métricas (Tabela 5) para DX (ID10, ID19, ID29, ID42, ID50, ID66, ID74, ID78, ID97, ID137, ID139, ID201, ID207, ID208, ID211, ID213, ID214, ID219, ID222, ID225, ID277, ID278, ID279, ID299, ID307, ID370, ID394, ID396, ID427, ID462, ID489, ID490, ID559, ID652, ID733, ID846, ID914, ID919, ID949).

**Tabela 5. Métricas de DX**

Métrica	Descrição	Finalidade para DX	Ocorrências	Ferramentas
Grau de satisfação	Mede o grau de satisfação de um desenvolvedor quanto ao consumo de um produto de <i>software</i>	Identificar potencial motivo de evasão	ID10, ID66, ID97, ID201, ID207, ID208, ID211, ID213, ID214, ID219, ID222, ID225	Acompanhamento e observação da comunidade ( <i>GitHub Issues, Twitter, Reddit</i> ); <i>user research</i> , ciclos de <i>feedback</i> ; <i>customer experience</i>

<b>Métrica</b>	<b>Descrição</b>	<b>Finalidade para DX</b>	<b>Ocorrências</b>	<b>Ferramentas</b>
Nível de frustração/felicidade	Avalia o quão frustrado ou feliz é a experiência de uso de um produto	Indicador sobre curva de aprendizagem	ID78, ID148	Acompanhamento e observação da comunidade ( <i>GitHub Issues, Twitter, Reddit</i> ); <i>user research</i> , ciclos de <i>feedback</i> ; <i>customer experience</i>
Tempo de <i>onboarding</i>	O tempo da chegada de um novo membro a um time de desenvolvimento até a completude de sua introdução ao modus operandi da organização e do time	Medir o tempo e complexidade de compreender processos para melhorias	ID427	Ciclos de <i>feedback</i> ; <i>onboarding</i>
Tempo para primeiro <i>commit</i>	O tempo da chegada de um novo membro a um time de desenvolvimento até a realização do seu primeiro <i>commit</i> para o repositório de código-fonte	Medir quanto tempo leva entre chegada e entrega, como indicador de facilidade de contribuir com o produto	ID277	<i>Onboarding</i>
Taxa de rotatividade	Índice percentual sobre a relação entre admissões e demissões versus o número de colaboradores de uma organização	Estratégias de gestão de pessoas	ID370	-
Evasão de usuários	Índice de usuários que deixam de utilizar uma ferramenta	Identificar presença de mercado	ID394, ID652	Portal do desenvolvedor ( <i>Backstage.io</i> )

<b>Métrica</b>	<b>Descrição</b>	<b>Finalidade para DX</b>	<b>Ocorrências</b>	<b>Ferramentas</b>
<i>Ranking</i> de fatores que levam a evasão	Classificação dos fatores que levaram a evasão de usuários de um produto	Identificar fragilidades no produto	ID222	Acompanhamento e observação da comunidade ( <i>GitHub Issues</i> , <i>Twitter</i> , <i>Reddit</i> ); <i>user research</i> , ciclos de <i>feedback</i> ; <i>customer experience</i>
Tempo de <i>build</i> e CI/CD	Tempo de compilação e tempo de execução de <i>pipeline</i> de integração contínua e entrega contínua	Tornar o processo de desenvolvimento mais ágil	ID10, ID19, ID74, ID299	<i>Gitlab CI</i> ; <i>Jenkins CI</i> ; <i>Circle CI</i> ; <i>Travis CI</i> ; <i>GitHub Actions</i>
Frequência de implantação	Relação entre o número de releases sobre um intervalo de tempo	Entender o número de vezes que novas funcionalidades ou melhorias são entregues aos clientes	ID10, ID949	<i>Gitlab CI</i> ; <i>Jenkins CI</i> ; <i>Circle CI</i> ; <i>Travis CI</i> ; <i>GitHub Actions</i> ; controle de versionamento
Tempo para produção	Tempo do início do desenvolvimento de uma funcionalidade até sua release	Melhorar estimativas	ID10, ID277, ID427	-
Tempo de vida de <i>pull request</i>	Tempo de vida da criação de um <i>pull request</i> até sua finalização	Melhorar processo de revisão de código visando <i>feedback</i> mais rápido para os desenvolvedores	ID490, ID949	-

<b>Métrica</b>	<b>Descrição</b>	<b>Finalidade para DX</b>	<b>Ocorrências</b>	<b>Ferramentas</b>
Tempo de integração <i>third-party</i>	Tempo de integração com aplicações/ferramentas de organizações terceiras	Verificar o processo de integração com o ecossistema em torno do produto	ID919	-
Tempo de integração com entregas de outros times	Tempo de integração com entregas de outros times sobre o mesmo produto	Verificar a dificuldade, baseada no tempo, de integrações de código entre equipes	ID370	-
Tempo de <i>setup</i> de API	Tempo de configuração inicial até primeira requisição para uma API	Compreender a morosidade para uso do produto, que pode levar a evasão	ID307	Portal do desenvolvedor ( <i>Backstage.io</i> )
Taxa de sucesso de implantação	Relação entre o número de releases implantadas com sucesso sobre o total de tentativas de implantação	Verificar pontos de falha no processo de implantação	ID949	<i>Gitlab CI;</i> <i>Jenkins CI;</i> <i>Circle CI;</i> <i>Travis CI;</i> <i>GitHub Actions</i>
Quantidade de <i>bugs/issues</i>	Quantidade de <i>bugs</i> ou <i>issues</i> identificados para um produto	Verificar qualidade das entregas para o cliente final e o interesse da comunidade para colaborar com a qualidade do produto	ID29, ID214, ID299	<i>Service Desk,</i> <i>GitHub Issues</i>
<i>North Star</i>	<i>Framework</i> utilizado para identificar um indicador utilizado como termômetro de assertividade de uma ferramenta	Guiar o time sobre uma proposta de valor a ser alcançada e/ou mantida	ID66, ID846	-



Métrica	Descrição	Finalidade para DX	Ocorrências	Ferramentas
OKR	<i>Framework</i> utilizado para identificar resultados que quando alcançados indicam a realização de um objetivo de negócio para o produto	Termômetro sobre a realização de objetivos com base no valor de negócio existente no produto	ID139, ID277	-
Tráfego	Dados analíticos sobre tráfego em canais de comunicação com desenvolvedores	Identificar canais e conteúdos mais utilizados	ID50, ID137, ID396, ID489	<i>Logging e Analytics (Hotjar, Mixpanel, Grafana)</i>
<i>Ranking</i> de tendências na comunidade	Classificação dos tópicos mais comentados pela comunidade nos diferentes canais existentes entre o produto e a comunidade e de tendências sobre produtos	Compreender as necessidades de mercado	ID50, ID97, ID137, ID214, ID222, ID225, ID279, ID396, ID489, ID914	Acompanhamento e observação da comunidade ( <i>GitHub Issues, Twitter, Reddit, Stack Overflow</i> )
Metadados de repositórios	Métricas diversas podem ser capturadas nos metadados de repositórios de código-fonte	Extrair dados técnicos sobre o status quo da comunidade	ID559	Controle de Versionamento

Existem 12 ocorrências que abordam o grau de satisfação de um usuário com um produto ponto de vista de DX externa, como consumidor do produto (ID10, ID66, ID97, ID201, ID207, ID208, ID211, ID213, ID214, ID219, ID222, ID225). De encontro, existem duas ocorrências indicando métricas a partir da frustração ou felicidade de um desenvolvedor: uma sobre o consumo (ID78) e outra sobre o desenvolvimento de um *software* (ID148).

Esses resultados indicam que uma possível, e importante métrica, é a avaliação qualitativa sobre o uso de um produto. Apesar da solidez de uma ferramenta, fatores não relacionados com o viés técnico impactam diretamente na experiência de consumo. Experiência não é somente sobre o produto entregar o que é esperado, mas também em toda a jornada que seu consumidor enfrenta durante seu uso. O mapeamento qualitativo dá indícios sobre fraquezas e forças existentes na presença de mercado, quantificando esses resultados para tomada de decisões.

Uma ocorrência abordou o tempo de *onboarding* de um recém chegado como uma métrica a ser observada (ID427). O tempo para o primeiro *commit* também foi citado em uma ocorrência (ID277). Esses resultados abordam o aspecto interno de DX, desde a chegada de um novo desenvolvedor à equipe até as entregas por ele realizadas.

Ainda sobre o aspecto interno e humano, a taxa de rotatividade de desenvolvedores pode ser observada como uma métrica para DX, sendo citada em um texto (ID370). Essa métrica, além de um termômetro para as equipes de DX e recursos humanos, também é um fator aliado na construção de marca empregadora, importante para atração e retenção de novos talentos.

A evasão também é observada para clientes (ID394, ID652). A primeira ocorrência trata da evasão ainda na etapa de descoberta do produto, quando o usuário não passa pela retenção e abandona o produto antes mesmo das primeiras requisições ou testes (ID394). A segunda, por outro lado, aborda o abandono por meio da contabilização de usuários ativos e inativos (ID652). Essas métricas podem ser associadas a outra, o *ranking* entre fatores que levam a evasão, que é abordado em uma ocorrência (ID222). Considerando a organização como um *player* do mercado, é de suma importância mapear os fatores que influenciam a perda de competitividade com relação a concorrentes.

Tempo é uma variável vista em diversas ocorrências, mas aplicado a diferentes pontos de vista. Quanto aos aspectos de DevOps, temos o tempo de *build* e de CI/CD presentes em quatro publicações (ID10, ID19, ID74, ID299). Além de tempo, ainda no aspecto DevOps, a frequência de implantação é outra métrica identificada, presente em duas ocorrências (ID10, ID949). Três citam o tempo para produção (ID10, ID277, ID427), enquanto duas citam o tempo de vida de um *pull request* (ID490, ID949).

O tempo de integração com produtos terceiros é observado em uma ocorrência (ID919). Por sua vez, o tempo de integração com entregas de outros times também é citado em apenas um texto (ID370). Também é abordada em apenas uma publicação o tempo de *setup* de uma API (ID307). Na atualidade, o tempo é um ativo valioso e essa visão se perpetua na cultura atual, e pode ser observada pelo senso comum. Com o advento da Cultura Ágil, as organizações de tecnologia buscam cada vez mais entregar mais qualidade e mais funcionalidades em um menor espaço de tempo. A entrega ágil depende também dos cenários de agilidade internos às rotinas de desenvolvimento.

Sucesso é observado em dois cenários. O primeiro é em relação a taxa de sucesso de implantação, presente em uma publicação (ID949). O segundo, em relação a quantidade de *bugs* e *issues* de uma ferramenta, que corresponde a três publicações (ID29, ID214, ID299). Sucesso é visto como atributo de credibilidade, quanto menos erros sua aplicação possui, mais bem quista no mercado ela é. Ele colabora na promoção de *marketing* de confiabilidade, potencialmente aumentando o número de usuários.

Métricas conhecidas no campo de produtos são também identificadas em quatro menções (ID66, ID139, ID277, ID846). A Métrica *North Star* (MNS, comumente chamada de Métrica da Estrela Guia, em português) é citada em dois textos (ID66, ID846). Ela é um *framework* utilizado para identificar uma métrica a ser utilizada como termômetro do negócio, e deve apoiar o principal valor entregue pelo seu produto para o seu cliente. OKR (*Objectives and Key Results*, Objetivos e Resultados Chave em português) é mencionado em dois textos (ID139, ID277). OKR é uma métrica utilizada para estabelecer resultados esperados que, quando satisfeitos, indicam a chegada a um objetivo. OKRs e MNSs podem ser estabelecidas com base na experimentação e observação do *status quo* do relacionamento produto *versus* usuário. Para tanto, é importante para a equipe responsável por DX estabelecer mecanismos de aproximação com a comunidade para que possam utilizá-las com êxito.

Muitas organizações estabelecem canais para contato com a comunidade em torno de seu produto, como envio de *newsletters* por e-mail, portal do desenvolvedor, FAQ e outros. Dados analíticos, como tráfego, quantidade de visualizações de *newsletter* ou de uma página de documentação estão presentes em quatro publicações (ID50, ID137, ID396, ID489). Eles podem ser coletados e utilizados como suporte para a tomada de decisão, além da compreensão sobre o ecossistema em que se insere.

Observar o comportamento da comunidade leva a compreensão dos caminhos do mercado, existem 10 ocorrências abordando este ponto (ID50, ID97, ID137, ID214, ID222, ID225, ID279, ID396, ID489, ID914). Por meio do *GitHub Issues*, *Stack Overflow*, *Twitter* e outros, podemos observar tendências da comunidade. Sejam com base nos pacotes mais baixados em repositórios ou em tópicos mais falados nos canais de contato com a comunidade. Tendências são métricas de ranqueamento que levam a compreender o que está em alta quanto ao seu produto. Além disso, o mapeamento de causas de evasão podem indicar uma tendência de comportamento. Essa informação pode ser utilizada como estratégia de mitigação de riscos de efeito manada para evasão.

Dados sobre comportamento da comunidade podem ser retirados de metadados de repositórios de código-fonte (ID559). Esses dados podem ser quantificados e aliados aos dados coletados pelo tópico abordado no parágrafo anterior. Permitindo, deste modo, a identificação do cenário em termos quantitativos e qualitativos. Usuários consomem produtos e serviços, e as organizações devem seguir a tendência do mercado para manterem-se presentes e competitivas nele.

É possível notar que métricas para DX não abarcam um rol específico, mas sim, um conjunto formado por métricas em diferentes aspectos e visões. Elas medem o comportamento de consumo e são utilizadas como sustentação da proposta de valor de um *software*. Seja no aspecto externo de DX, sobre o consumo, quanto no aspecto interno, sobre produção de software, no aspecto de cultura organizacional. Métricas não representam apenas números, mas também uma classificação em termos qualitativos. Elas não se encontram dissociadas e atuam de forma complementar uma à outra. Podem ser coletadas por ferramentas, identificadas neste estudo GLR, e os agentes, em geral, são as equipes responsáveis por DX na organização.

#### **4. Conclusões**

Esta pesquisa buscou evidenciar a percepção sobre DX na perspectiva de profissionais da indústria. Elencamos a sua definição, ferramentas de apoio e métricas, com a finalidade de compreender a prática de DX como meio de apoio a comunidade acadêmica e profissional. Para tanto, conduzimos uma abordagem de *Grey Literature Review* para identificar em postagens de *blogs* o que é compartilhado sobre o objeto de pesquisa, pelos seus profissionais.

Além dos impactos da pandemia do COVID-19, é identificado um crescimento da comunidade global de desenvolvedores entre 2020 e 2022. Analisamos as publicações nesse recorte temporal e confrontamos os resultados da definição de DX com o conceito moldado na academia. O conceito gerado a partir da nossa análise engloba DX como práticas de UX e CX. Associadas, elas provêm uma boa experiência internamente (etapas de desenvolvimento) e externamente (consumo de artefatos). DX se perpetua nos ambientes *técnico* e *social*, envolve aspectos nos domínios de *cultura*,

*ferramentas, processos/técnicas e pessoas*. Gera valor agregado a um produto por meio das relações destes com a comunidade em que está inserido.

A comunidade e a documentação são as principais ferramentas para DX, a partir do que foi observado. Evidencia-se que a comunidade é a principal fonte de informações sobre o cenário de DX sobre o produto. Por meio dela, é possível obter informações sobre tendências, dificuldades e mapear a efetividade da estratégia empregada para melhoria na experiência. Por ela, se realiza a propagação da marca no ecossistema. Já a documentação é observada como o guia técnico, sendo o principal suporte para a relação individual entre desenvolvedores e *software*.

Foi possível evidenciar que o mercado carece de ferramentas especializadas para análise de DX. Ou ainda, metodologias para implantação, como meio de propagação da cultura de DX nas indústrias. A observação da comunidade, como mencionado, é a principal ferramenta para obter dados sobre DX. Ela se torna um termômetro sobre a efetividade da estratégia empregada em torno do produto. A satisfação de uso, o que envolve a experiência, é observada como a principal métrica. Os principais meios por onde a observação ocorre é através dos canais de comunicação estabelecidos pela organização e pela comunidade. É observada uma grande popularidade de alguns destes: *GitHub Issues, Twitter, Reddit e Stack Overflow*.

DX existe desde a criação do primeiro *software*, dado que a experiência precede a definição do termo, que é mais recente. Por ser uma área emergente, nem todas as organizações delegam a responsabilidade a um papel ou área. Todavia, muitas organizações caminham para a formação de uma área na organização responsável por DX. Essa área transita entre aspectos técnicos e comerciais, provendo DX na ótica interna dos trabalhadores da organização e externa por meio de seus clientes.

#### **4.1. Trabalhos Futuros**

Nossa pesquisa avaliou a perspectiva de profissionais da indústria, primariamente, no que tange a revisão literária. Uma limitação é em relação a fonte de dados, GL abrange uma gama de meios pelos quais o conhecimento pode ser propagado. Sugere-se a extensão para outras fontes, complementando a visão profissional da fonte analisada.

De encontro, a pesquisa buscou evidenciar os aspectos de DX em um intervalo de tempo no qual mudanças disruptivas ocorreram. A pandemia do COVID-19 mudou muito sobre o modo como vivemos, inclusive como experienciamos a engenharia de *software*. A computação trouxe, para o mundo, uma inovação exponencial, assim como as ocorridas nas Revoluções Industriais. Uma abordagem possível seria a realização de um estudo buscando avaliar DX na perspectiva da indústria ao longo do tempo, caracterizando a evolução da experiência conforme a evolução da computação.

Por fim, trabalhos futuros podem explorar questões não respondidas por esta pesquisa. Por exemplo, a aderência das organizações para a criação de áreas responsáveis pela DX de seus produtos. Ou ainda, avaliar a presença destes papéis no Brasil e no mundo, assim como os fatores que advogam sobre práticas de DX.

#### **5. Agradecimentos**

Agradeço a Faculdade de Computação da Universidade Federal de Mato Grosso do Sul, seus professores, técnicos e alunos. Agradeço ainda aos professores e alunos do

LEDES/FACOM (Laboratório de Desenvolvimento e Engenharia de *Software*), em especial ao meu orientador Prof. Dr. Awdren de Lima Fontão e os colegas que colaboraram para a realização desta pesquisa, Ana Elisa da Silva Cunha e Carlitos Fioravante Vieira de Oliveira. Agradeço também a minha família, amigos e colegas que me acompanharam e me incentivaram a chegar até aqui.

## Referências

- Basili, V. (1994) “GQM approach has evolved to include models”, *IEEE SOFTWARE*, vol. 11, n. 1, p. 8-8.
- Brito, G., Terra, R., e Valente, M. T. (2018) “Monorepos: a multivocal literature review”. arXiv preprint arXiv:1810.09477.
- Cruzes, D. S. e Dyba, T. (2011) “Recommended steps for thematic synthesis in software engineering”, In: 2011 International Symposium on Empirical Software Engineering and Measurement, IEEE, p. 275-284.
- Constantino, K., Souza, M., Zhou, S., Figueiredo, E., e Kästner, C. (2023) “Perceptions of open-source software developers on collaborations: An interview and survey study”, *Journal of Software: Evolution and Process*, vol. 35, n. 5.
- Dewan, P. (2015) “Towards emotion-based collaborative software engineering”, In: 015 IEEE/ACM 8th International Workshop on Cooperative and Human Aspects of Software Engineering, p. 109-112.
- Fagerholm, F. e Münch, J. (2012) “Developer experience: Concept and definition”, In: 2012 international conference on software and system process (ICSSP), IEEE, p. 73-77.
- Fagerholm, F. (2015) “Software developer experience: Case studies in lean-agile and open source environments”, (Tese de Doutorado) University of Helsinki.
- Fontão, A., Ekwoje, O. M., Santos, R. e Dias-Neto, A. C. (2017) “Facing up the primary emotions in Mobile Software Ecosystems from Developer Experience”, In: Proceedings of the 2nd workshop on social, human, and economic aspects of software, p. 5-11.
- Fontão, A., Cleger-Tamayo, S., Wiese, I., Pereira dos Santos, R., e Dias-Neto, A. C. (2020) “On value creation in developer relations (devrel) a practitioners’ perspective”, In: Proceedings of the 15th International Conference on Global Software Engineering, p. 33–42.
- \_\_\_\_\_ (2023) “A Developer Relations (DevRel) model to govern developers in Software Ecosystems”, *Journal of Software: Evolution and Process*, vol. 35, n. 5.
- Ford, D., Storey, M.-A., Zimmermann, T., Bird, C., Jaffe, S., Maddila, C., Butler, J. L., Houck, B. e Nagappan, N. (2022) “A tale of two cities: Software developers working from home during the covid-19 pandemic”, *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol. 31, n. 2, p. 1-37.
- Garousi, V., Felderer, M. e Mäntylä, M. V. (2019). “Guidelines for including grey literature and conducting multivocal literature reviews in software engineering”, *Information and software technology*, vol. 106, p. 101-121.

- GitHub (2020) The 2020 State of Octoverse, <https://octoverse.github.com/2020/>, Dezembro.
- GitHub (2021). The 2021 State of Octoverse, <https://octoverse.github.com/2021/>, Novembro.
- \_\_\_\_\_ (2022). Octoverse 2022: The state of open source | The State of Octoverse, <https://octoverse.github.com>, Dezembro.
- Greiler, M., Storey, M. e Noda, A. (2023) “An Actionable Framework for Understanding and Improving Developer Experience”, *IEEE Transactions on Software Engineering*, vol. 49, n. 4, p. 1411-1425.
- Jamalpur, S., Navya, Y. S., Raja, P., Tagore, G., e Rao, G. R. K. (2018). “Dynamic malware analysis using cuckoo sandbox”, In: 2018 Second international conference on inventive communication and computational technologies (ICICCT), p. 1056-1060.
- Kamei, F., Wiese, I., Pinto, G., Ribeiro, M. e Soares, S. (2020) “On the Use of Grey Literature: A Survey with the Brazilian Software Engineering Research Community”, In: *Proceedings of the 34th Brazilian symposium on Software Engineering*, p. 183-192.
- Kamei, F., Wiese, I., Pinto, G., Ferreira, W., Ribeiro, M., Souza, R. e Soares, S. (2022) “Assessing the Credibility of Grey Literature: A Study with Brazilian Software Engineering Researchers”, *Journal of Software Engineering Research and Development*, vol. 10, p. 9:1 - 9:20.
- Keenan, R., Kersten, N. e O’Connell, C. (2023). 2023 State of DevOps Report, <https://www.puppet.com/resources/history-of-devops-reports>, Janeiro.
- Khan, M. S., Khan, A. W., Khan, F., Khan, M. A. e Whangbo, T. K. (2022) “Critical challenges to adopt DevOps culture in software organizations: a systematic review”, *IEEE Access*, vol. 10, 14339-14349.
- Kuusinen, K., Petrie, H., Fagerholm, F., e Mikkonen, T. (2016). “Flow, Intrinsic Motivation, and Developer Experience in Software Engineering”, In: *17th International Conference, XP 2016 Edinburgh, UK, May 24–27, 2016 Proceedings*, p. 104–117.
- Leite, L., Rocha, C., Kon, F., Milojevic, D. e Meirelles, P. (2019) “A survey of DevOps concepts and challenges”, *ACM Computing Surveys (CSUR)*, vol. 52, n. 6, p. 1-35.
- Londero, R. R. (2017) “Ainda AIDA? - Uma revisão histórico-crítica da hierarquia dos efeitos publicitários”, *Animus, Revista Interamericana De Comunicação Midiática*, vol. 16, n. 31.
- Manikas, K. (2016) “Revisiting software ecosystems Research: A longitudinal literature study”, *J. Syst. Softw.*, vol. 117, pp. 84–103.
- Menezes, G., Braga, W., Fontão, A., Hora, A. e Cafeo, B. (2022) “Assessing the Impact of Code Samples Evolution on Developers’ Questions”, In: *Proceedings of the XXXVI Brazilian Symposium on Software Engineering*, p. 321-330.
- Neto, P. A. M. S., Mannan, U. A., Almeida, E. S., Nagappan, N., Lo, D., Kochhar, P. S., Gao, C. e Ahmed, I. (2021) “A deep dive into the impact of covid-19 on software

- development”, IEEE Transactions on Software Engineering, vol. 48, n.9, p. 3342-3360.
- Nylund, A. (2020) “A multivocal literature review on developer experience”, (Dissertação de Mestrado) Aalto University.
- Novielli, N. e Serebrenik, A. (2019) “Sentiment and emotion in software engineering”, IEEE Software, vol. 36, n. 5, p. 6-23.
- Oliveira, R., Ajala, C., Viana, D., Cafeo, B., e Fontão, A. (2021). “Developer relations (devrel) roles: An exploratory study on practitioners’ opinions”, In: Proceedings of the XXXV Brazilian Symposium on Software Engineering, p. 363-367.
- Potrich, L. N. e Selig P. M. (2019) “Riscos e impactos da perda de conhecimento no setor de tecnologia: Relação com fatores humanos”, In: Anais do Congresso Internacional de Conhecimento e Inovação—ciki, vol. 1, n. 1.
- Ragkhitwetsagul, C., Krinke, J., Paixao, M., Bianco, G., & Oliveto, R. (2019) “Toxic code snippets on stack overflow”, IEEE Transactions on Software Engineering, vol. 47, n. 3, p. 560-581.
- Storey, M.-A., Singer, L., Cleary, B., Filho, F. F., e Za-galsky, A. (2014). “The (r) evolution of social media in software engineering”, In: Proceedings of the on Future of Software Engineering, FOSE ’14.
- Uddin, G., Alam, O. e Serebrenik, A. (2022) “A qualitative study of developers’ discussions of their problems and joys during the early COVID-19 months”, Empirical Software Engineering, vol. 27.
- Williams, A. e Rainer, A. (2017) “Toward the use of blog articles as a source of evidence for software engineering research”, In: Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering, EASE’17, p. 280–285.
- Williams, A. (2019) “Finding high-quality grey literature for use as evidence in software engineering research”, (Tese de Doutorado) University of Canterbury.
- Zaid, S., w Patwayati, P. (2021) “Impact of customer experience and customer engagement on satisfaction and loyalty: A case study in Indonesia”, The Journal of Asian Finance, Economics and Business, vol. 8, n. 4, p. 983-992.