

UNIVERSIDADE ESTADUAL DE MARINGÁ  
CENTRO DE TECNOLOGIA  
DEPARTAMENTO DE INFORMÁTICA  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

ANDERSON BERGAMINI DE NEIRA

**Estudo exploratório sobre o perfil de contribuição dos  
desenvolvedores da plataforma Topcoder**

Maringá

2018

ANDERSON BERGAMINI DE NEIRA

**Estudo exploratório sobre o perfil de contribuição dos desenvolvedores da plataforma Topcoder**

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Departamento de Informática, Centro de Tecnologia da Universidade Estadual de Maringá, como requisito parcial para obtenção do título de Mestre em Ciência da Computação.

Orientador: Prof. Dr. Igor Fábio Steinmacher

Coorientador: Prof. Dr. Edson Alves de Oliveira Junior

Maringá  
2018

Dados Internacionais de Catalogação-na-Publicação (CIP)  
(Biblioteca Central - UEM, Maringá – PR, Brasil)

N415e Neira, Anderson Bergamini de  
Estudo exploratório sobre o perfil de  
contribuição dos desenvolvedores da plataforma  
Topcoder / Anderson Bergamini de Neira. -- Maringá,  
PR, 2018.  
97 f.: il. color.

Orientador: Prof. Dr. Igor Fábio Steinmacher.  
Coorientador: Prof. Dr. Edson Alves de Oliveira  
Junior.

Dissertação (mestrado) - Universidade Estadual de  
Maringá, Centro de Tecnologia, Departamento de  
Informática, Programa de Pós-Graduação em Ciência da  
Computação, 2018.

1. Desenvolvimento de software. 2. Crowdsourcing.  
3. Plataforma Topcoder. I. Steinmacher, Igor Fábio,  
orient. II. Oliveira Junior, Edson Alves de, orient.  
III. Universidade Estadual de Maringá. Centro de  
Tecnologia. Departamento de Informática. Programa de  
Pós-Graduação em Ciência da Computação. IV. Título.

CDD 23.ed. 005.12

Márcia Regina Paiva de Brito – CRB-9/1267

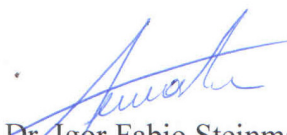
## FOLHA DE APROVAÇÃO

ANDERSON BERGAMINI DE NEIRA

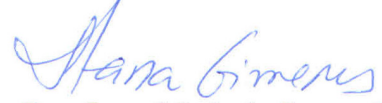
### **Estudo exploratório sobre o perfil de contribuição dos desenvolvedores da plataforma Topcoder**

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Departamento de Informática, Centro de Tecnologia da Universidade Estadual de Maringá, como requisito parcial para obtenção do título de Mestre em Ciência da Computação pela Banca Examinadora composta pelos membros:

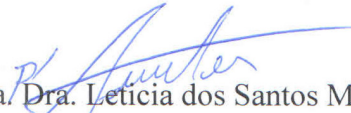
#### BANCA EXAMINADORA



Prof. Dr. Igor Fabio Steinmacher  
Universidade Tecnológica Federal do Paraná – DACOM/UTFPR-CM



Profa. Dra. Itana Maria de Souza Gimenes  
Universidade Estadual de Maringá – DIN/UEM



Profa. Dra. Leticia dos Santos Machado  
Universidade do Vale do Rio dos Sinos – UNISINOS  
participação por videoconferência

Aprovada em: 21 de junho de 2018.

Local da defesa: Sala 120, Bloco C56, *campus* da Universidade Estadual de Maringá.

# Estudo exploratório sobre o perfil de contribuição dos desenvolvedores da plataforma Topcoder

## RESUMO

O *crowdsourcing* é um movimento que tem atraído empresas de renome e participantes do mundo inteiro. Atualmente, existem diversas plataformas de *crowdsourcing* focadas em desenvolvimento de software. A plataforma Topcoder é uma das populares do mercado. Dado o crescimento deste modelo de desenvolvimento, pesquisadores têm estudado formas de melhorar a interação entre solicitante, usuário e plataforma, com o objetivo de garantir uma experiência agradável para todos. Um fator que afeta diretamente a plataforma e as empresas solicitantes é o perfil de contribuição dos participantes, pois são eles quem solucionam as demandas criadas pelos clientes. Assim, o objetivo deste trabalho é investigar o perfil de contribuição dos usuários que participam do desenvolvimento de software na plataforma Topcoder. Para analisar o perfil de contribuição dos usuários, este estudo foi dividido em duas fases: análise da hiperespecialização; e análise do perfil de contribuição. Para as duas fases foram utilizados dados coletados da plataforma Topcoder por meio de sua API pública. As análises conduzidas foram quantitativas, baseadas em estatística descritiva e testes de comparação entre populações. Na primeira fase foi investigado como é possível identificar os usuários que contribuem majoritariamente em desafios com a mesma classificação (*development*, *design* ou *data science*), ou em desafios que utilizam uma mesma tecnologia, sendo classificados, então, como hiperespecialistas. Como resultado da primeira fase, foi possível verificar a existência de usuários hiperespecialistas em termos de tipos de desafio e tecnologias requeridas. Apesar deste perfil aparecer, ele não é o mais vitorioso. Na segunda fase (análise do perfil de contribuição), o objetivo foi o de identificar diferentes perfis de contribuição baseado nas participações dos usuários nos desafios. Para analisar como os usuários da plataforma contribuem, um conjunto de atributos relacionados às suas participações foram coletados, curados e utilizados para agrupar os usuários utilizando o algoritmo *Model-based Clustering*. Na segunda fase, observaram-se cinco grupos distintos de usuários: elite, recém-chegados, desistentes, tardios e usuários aventureiros. Dentre os grupos é possível verificar usuários altamente engajados com a plataforma, além de usuários que estão há menos tempo na plataforma e têm dificuldade em evoluir. Acredita-se que, de posse desses resultados seja possível evoluir na compreensão das ações dos participantes de *crowdsourcing*, contribuindo para a manutenibilidade do ecossistema.

**Palavras-chave:** *crowdsourcing*, Topcoder, perfil de contribuição, hiperespecialização.

# Exploratory study about contribution profiles of Topcoder developers

## *ABSTRACT*

Crowdsourcing is a movement that has attracted renowned companies and participants from around the world. Currently, there are several crowdsourcing platforms focused on software development. The Topcoder platform is one of the popular platforms on the market. Given the growth of this development model, researchers have been studying ways to improve the interaction between requester, user and platform, with the goal of ensuring a pleasant experience for all. One factor that directly affects the platform and the requesting companies is the contribution profile of the participants, since they are the ones who solve the demands created by the clients. Thus, the objective of this work is to investigate the contribution profile of users who participate in software development in the Topcoder platform. In order to analyze the contribution profile of the users, this study was divided in two phases: hyperspecialization analysis; and contribution profile analysis. In the first phase, we investigated this phenomenon by considering the hyperspecialization as something related to the challenges that developers participated (development, design or data science); and to the technologies that are required to accomplish the challenges. The analysis of hyperspecialization was quantitative and was conducted in two steps: in the first step we made use of data from all types of challenges from an 18-month time window; the second focuses only on development challenges and encompasses all the contributions of the most participatory users. As a result of the first phase, it was possible to verify the existence of hyperespecialistas users, although this profile is not the most victorious. In the second phase (contribution profile analysis), the objective was to identify different contribution profiles based on users' participation. We collected different attributes related to user participation and we grouped users using the Model-based Clustering algorithm. We observed five distinct groups of users: experts, newcomers, dropouts, late users and adventurous users. Among the groups it is possible to check users highly engaged with the platform, in addition to users who are less time on the platform and have difficulty in evolving. We believe that, in the possession of these results, it is possible to evolve in understanding the actions of crowdsourcing participants, contributing to the maintenance of the ecosystem.

**Keywords:** crowdsourcing, Topcoder, contribution profile, hyperspecialization.

## LISTA DE FIGURAS

Figura - 1.1	Visão geral do estudo conduzido neste trabalho . . . . .	14
Figura - 2.1	<i>Crowdsourcing</i> e a Engenharia de Software (Mao et al., 2015a) . .	20
Figura - 2.2	Interação entre as camadas (Zou et al., 2014) . . . . .	21
Figura - 2.3	Soluções enviadas por usuários do <i>crowdsourcing</i> (Dwarakanath et al., 2016) . . . . .	23
Figura - 2.4	Processo de desenvolvimento de software utilizado pela plataforma Topcoder (Mao et al., 2015b) . . . . .	27
Figura - 3.1	Representação geral do método . . . . .	36
Figura - 3.2	Exemplo das conquistas de um usuário extraído da plataforma Topcoder . . . . .	41
Figura - 4.1	Visão de alto nível do método seguido na fase da Hiperespecialização	45
Figura - 4.2	Método de pesquisa para a Etapa 1 . . . . .	46
Figura - 4.3	Distribuição do número de desafios participados pelos usuários nos períodos analisados . . . . .	49
Figura - 4.4	Distribuição das participações nos períodos verificados . . . . .	50
Figura - 4.5	Representação gráfica do método utilizado na Etapa 2 da análise da hiperespecialização . . . . .	52
Figura - 4.6	Número de hiperespecialistas por tecnologia ou combinação de tecnologias . . . . .	56
Figura - 4.7	Distribuição de diferentes tecnologias que os usuários tiveram contato. . . . .	59
Figura - 4.8	Número de desafios por usuário: hiperespecialistas versus não especialistas . . . . .	60
Figura - 4.9	Número de vitórias por usuário: hiperespecialistas vs. não especialistas . . . . .	62
Figura - 4.10	Relação de vitória (vitória/desafio): comparando hiperespecialistas e não especialistas . . . . .	63
Figura - 5.1	Representação gráfica do método utilizado na fase de análise da constituição do perfil de contribuição . . . . .	67
Figura - 5.2	Distribuição dos usuários entre os grupos identificados . . . . .	70
Figura - 5.3	Distribuição dos usuários na característica dias na plataforma . .	77

Figura - 5.4	Distribuição dos usuários na característica total de dias para a primeira submissão . . . . .	78
Figura - 5.5	Distribuição dos usuários na característica total de dias em competição . . . . .	79
Figura - 5.6	Distribuição dos usuários na característica total de conquistas . .	79
Figura - 5.7	Distribuição dos usuários na característica total de desafios com submissão . . . . .	80
Figura - 5.8	Distribuição dos usuários na característica total de vitórias . . . .	81
Figura - 5.9	Distribuição dos usuários na característica total de tecnologias . .	81
Figura - 5.10	Distribuição dos usuários na característica total de habilidades . .	82
Figura - 5.11	Distribuição dos usuários na característica hiperespecialista . . . .	83
Figura - 2.1	Digrama de entidade e relacionamento do banco de dados . . . . .	98



## LISTA DE TABELAS

Tabela - 2.1	Plataformas de <i>crowdsourcing</i> para o desenvolvimento de software (Mao et al., 2015a) . . . . .	25
Tabela - 3.1	Relacionamentos da tabela “Membro” . . . . .	40
Tabela - 4.1	Distribuição dos hiperespecialistas pelo número de tecnologias nas quais são especializados . . . . .	55
Tabela - 4.2	Top-10 tecnologias em termos do número de hiperespecialistas disponíveis . . . . .	55
Tabela - 4.3	Top-10 tecnologias sem hiperespecialista . . . . .	57
Tabela - 4.4	Número de hiperespecialista por desafio . . . . .	58
Tabela - 5.1	Centro das característica nos grupos . . . . .	70
Tabela - 5.2	Relação das abreviaturas com os nomes completos das características . . . . .	71
Tabela - 5.3	As dez conquistas que os usuários do grupo 5 mais atingiram . . .	76
Tabela - 5.4	Proporção do tempo utilizado na plataforma para competir . . . .	78
Tabela - 1.1	Relação das abreviaturas com os nomes completos das características . . . . .	95
Tabela - 1.2	Valor mínimo para cada atributo dos grupos encontrados . . . . .	96
Tabela - 1.3	Valor máximo para cada atributo dos grupos encontrados . . . . .	96
Tabela - 1.4	Primeiro quartil para cada atributo dos grupos encontrados . . . .	96
Tabela - 1.5	Terceiro quartil para cada atributo dos grupos encontrados . . . .	96
Tabela - 1.6	Médias dos grupos encontrados . . . . .	96

## LISTA DE SIGLAS E ABREVIATURAS

- AMT:** *Amazon Mechanical Turk*
- API:** *Application Programming Interface*
- CSE:** *Crowdsourced Software Engineering*
- DER:** *Diagrama de entidade e relacionamento*
- HTML:** *HyperText Markup Language*
- OSS:** *Open Source Software*
- Q&A:** *Questions & Answers*
- QP:** *Questão de Pesquisa*
- SRM:** *Single Round Matches*
- UML:** *Unified Modeling Language*
- URL:** *Uniform Resource Locator*

# SUMÁRIO

<b>1</b>	<b>Introdução</b>	<b>12</b>
1.1	Contribuições . . . . .	16
1.2	Outras contribuições . . . . .	17
1.3	Artigos . . . . .	17
<b>2</b>	<b>Revisão de literatura</b>	<b>18</b>
2.1	Fundamentação teórica . . . . .	19
2.1.1	Contextualização do <i>crowdsourcing</i> . . . . .	19
2.1.2	Interação do modelo . . . . .	20
2.1.3	Vantagens e desvantagens . . . . .	21
2.1.4	Plataformas de <i>crowdsourcing</i> . . . . .	23
2.1.5	Topcoder . . . . .	25
2.2	Revisão bibliográfica . . . . .	28
2.2.1	Recomendação de participantes . . . . .	28
2.2.2	Motivação . . . . .	29
2.2.3	Recompensas e contribuições . . . . .	30
2.2.4	Perfil de contribuição dos participantes . . . . .	31
2.2.5	Novos usuários no <i>crowdsourcing</i> . . . . .	33
2.2.6	Considerações finais . . . . .	34
<b>3</b>	<b>Metodologia de pesquisa</b>	<b>35</b>
3.1	Hiperespecialização . . . . .	36
3.2	Perfil de contribuição . . . . .	36
3.3	Coleta dos dados . . . . .	37
3.4	Modelo dos dados . . . . .	39
<b>4</b>	<b>Fase da Hiperespecialização</b>	<b>44</b>
4.1	Etapa 1: Análise preliminar do perfil de hiperespecialista . . . . .	44
4.1.1	Método de pesquisa . . . . .	45
4.1.2	Filtragem e análise dos dados . . . . .	46
4.1.3	Resultados . . . . .	48
4.2	Etapa 2: Hiperespecialização em desafios de desenvolvimento . . . . .	50
4.2.1	Método de pesquisa . . . . .	50
4.2.2	Coleta e seleção de dados . . . . .	51
4.2.3	Análise de dados . . . . .	53

4.2.4	Resultados . . . . .	54
4.3	Implicações . . . . .	62
4.4	Limitações e Ameaças à Validade . . . . .	64
4.5	Conclusão . . . . .	65
<b>5</b>	<b>Perfil de contribuição</b>	<b>66</b>
5.1	Método de Pesquisa . . . . .	66
5.2	Filtragem e análise dos dados . . . . .	68
5.3	Resultados . . . . .	69
5.3.1	Grupo 1: Elite . . . . .	70
5.3.2	Grupo 2: Recém-chegados . . . . .	71
5.3.3	Grupo 3: Desistentes . . . . .	72
5.3.4	Grupo 4: Tardios . . . . .	73
5.3.5	Grupo 5: Aventureiros . . . . .	74
5.3.6	Comparação entre os grupos . . . . .	76
5.3.7	Implicações . . . . .	82
5.4	Limitações e ameaças a validade . . . . .	84
5.5	Conclusão . . . . .	85
<b>6</b>	<b>Conclusão</b>	<b>86</b>
6.1	Lições aprendidas . . . . .	87
6.2	Trabalhos futuros . . . . .	87
	<b>REFERÊNCIAS</b>	<b>89</b>
<b>A</b>	<b>Apêndice A</b>	<b>95</b>
A.1	Dados dos grupos . . . . .	95
<b>B</b>	<b>Apêndice B</b>	<b>97</b>
B.1	Banco de dados . . . . .	97

---

# Introdução

---

Um novo conceito para o desenvolvimento de software está se consolidando na indústria atraindo a atenção de empresas (Mao et al., 2016), desenvolvedores (Karim et al., 2016; Xie et al., 2014) e pesquisadores de todo o mundo. Mao et al. (2015a) definem o *crowdsourcing* para o desenvolvimento de software como um conjunto de demandas relacionadas à engenharia de software, que são disponibilizadas para um grupo potencialmente grande de pessoas que não possuem relações diretas com a empresa solicitante e essas pessoas decidem em quais demandas vão contribuir. As demandas são solicitadas por empresas que necessitam de artefatos relacionados a software como projetar telas, executar testes, corrigir *bugs*, executar tarefas de codificação, entre outros. O *crowdsourcing* para o desenvolvimento de software utiliza os desenvolvedores fisicamente distantes para realizar tarefas que são propostas por empresas do mundo inteiro.

O *crowdsourcing* proporciona aos participantes engajados uma forma de ganhar dinheiro, notoriedade e até mesmo oportunidades profissionais (Hossain, 2012). As empresas que solicitam as demandas por meio do *crowdsourcing* veem no modelo uma forma econômica e confiável de desenvolver software, contando com a “sabedoria das multidões” (Surowiecki, 2005) e a diversidade de experiências para completar as tarefas as solicitadas. Uma das maiores plataformas de *crowdsourcing* para o desenvolvimento de software do mundo é a Topcoder<sup>1</sup>, contando com mais de um milhão e duzentos mil participantes cadastrados<sup>2</sup>. O papel da plataforma é conectar as empresas com os desenvolvedores, visando ao desenvolvimento de artefatos de software de qualidade.

---

<sup>1</sup><http://www.topcoder.com>

<sup>2</sup><https://www.topcoder.com/community/members/>

Pesquisadores têm estudado vários assuntos relacionados ao *crowdsourcing*. Por exemplo, a área de recomendação de participantes para tarefas é o assunto tratado no trabalho de Ye e Wang (2016) que propõem um método para recomendar os participantes adequados para as tarefas. Saremi e Yang (2015) relatam que os membros mais experientes buscam tarefas de empresas de renome mundial e as tarefas que possuem um bom retorno econômico, porém Gadiraju et al. (2015) ressaltam que altas recompensas tendem a atrair pessoas com más intenções. Por fim, Malone et al. (2011) citam que alguns desenvolvedores podem ser classificados como hiperespecialistas, pois conhecem a fundo uma quantidade limitada da tecnologias.

Apesar do número de trabalhos relacionados à *crowdsourcing* ter aumentado nos últimos anos (Ambreen e Ikram, 2016; Mao et al., 2015a), existem tópicos que merecem mais atenção. Um deles é o perfil de contribuição dos participantes envolvidos com *crowdsourcing*. Conhecendo como agem os membros da plataforma, é possível verificar se os modelos empregados são adequados à realidade dos trabalhadores. O conhecimento do perfil de contribuição dos membros das plataformas de *crowdsourcing* também pode influenciar a decisão de escolha entre as plataformas existentes. Além disso, conhecendo o perfil de contribuição dos participantes, as empresas podem tomar decisões para melhor acomodar os diferentes tipos de participantes, visando cativar um público de desenvolvedores e atrair outros mais, para que o software entregue por meio do *crowdsourcing* possa superar as expectativas do cliente.

Tendo em vista a lacuna existente na literatura acerca do perfil dos participantes de *crowdsourcing* e a popularidade da plataforma Topcoder, o objetivo geral deste trabalho é conduzir um estudo exploratório baseado nas ações dos participantes da plataforma Topcoder para responder a seguinte questão de pesquisa (QP): *Como pode ser representado o perfil de contribuição dos usuários da plataforma Topcoder?* Com o intuito de alcançar o objetivo principal deste trabalho, dois objetivos específicos estão representados nas seguintes questões de pesquisa:

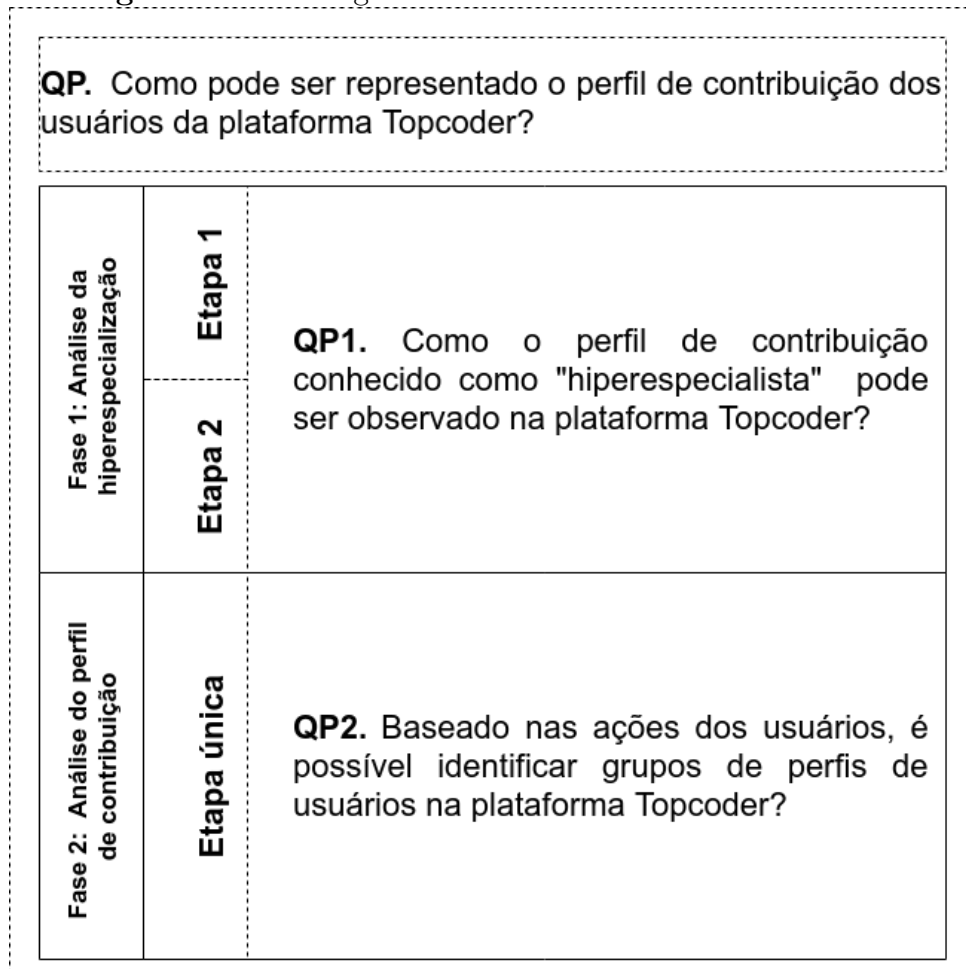
- **QP1.** Como o perfil de contribuição conhecido como “hiperespecialista” (Malone et al., 2011) pode ser observado na plataforma Topcoder?
- **QP2.** Baseado nas ações dos usuários, é possível identificar grupos de perfis de usuários na plataforma Topcoder?

Para atingir estes objetivos, foi desenvolvido um sistema que coleta informações relacionadas às ações dos membros da plataforma Topcoder. Essas informações estavam disponíveis nas *Application Programming Interfaces* (APIs) que a plataforma utiliza para

completar parte das páginas dos perfis dos usuários disponibilizadas no site oficial. Essas informações foram armazenadas em um banco de dados relacional.

De posse das informações coletadas, um estudo de duas fases foi realizado para responder às questões de pesquisa. A Figura - 1.1 demonstra a divisão do estudo entre as fases e qual é o objetivo específico de cada fase.

**Figura 1.1:** Visão geral do estudo conduzido neste trabalho



A primeira fase investiga uma característica específica para definir um perfil de contribuição, a hiperespecialização. Nesta fase foram executadas duas etapas distintas que se complementam. Na primeira etapa, o objetivo era abstrair uma visão geral de como se dá a manifestação do fenômeno da hiperespecialização. Desse modo, um grupo de 664 usuários foi aleatoriamente amostrado com o intuito de verificar se era possível encontrar usuários hiperespecialistas. O usuário só estaria apto a ser amostrado caso ele tivesse se inscrito em, ao menos, um desafio. Neste ponto, foram analisados todos os tipos de participação em desafios que os usuários possuíam. Como resultado, foi possível identificar

que muitos usuários param de competir muito cedo. Em contrapartida, os usuários que permaneceram competindo, em geral, continuaram competindo majoritariamente na mesma classificação de desafio.

Após os resultados da Etapa 1 indicarem ser possível encontrar esta característica formadora de perfil, outra etapa com 2.086 usuários foi conduzida. O intuito desta era verificar a existência da hiperespecialização em um contexto específico unicamente entre os usuários que mais participaram de desafios classificados como *desenvolvimento*, bem como extrair detalhes sobre o modo de contribuição. Nessa etapa, a seleção dos usuários foi diferente. Dentre os critérios escolhidos para selecionar os usuários é oportuno destacar que, para o usuário ser elegível, o mesmo deveria ter submetido soluções para três desafios de desenvolvimento distintos. Além disso, o desafio deveria requerer ao menos uma tecnologia, pois as tecnologias requeridas foram utilizadas para definir os usuários que seriam identificados como hiperespecialistas. Os resultados da Etapa 2 da Fase 1 indicam que 60% dos desenvolvedores são considerados hiperespecialistas. Também foi descoberto que existe uma alta correlação entre o número de desafios disponíveis com tecnologias e o número de especialistas atraídos pelos desafios. Portanto, as tecnologias mais exigidas no Topcoder, como Java, Javascript, .NET, HTML (HyperText Markup Language) e iOS também apresentam um alto número de especialistas identificados. Curiosamente, não foi possível identificar hiperespecialistas para tecnologias importantes como MySQL, PostgreSQL e Docker.

A segunda fase consistiu em identificar grupos de desenvolvedores com características similares baseados no perfil de contribuição na plataforma. Para isso foram elencadas nove características que podem definir o perfil de contribuição dos usuários. Todas as características estavam disponíveis na coleção de dados. O resultado da contagem dos atributos para cada usuário, juntamente com o algoritmo *Model-based Clustering*, foram utilizados para agrupar os usuários que possuíam perfis semelhantes. Os resultados apontaram cinco grupos distintos de usuários: elite, recém-chegados, desistentes, tardios e usuários aventureiros. Dentre os grupos, é oportuno destacar três. O primeiro trata-se de um grupo com alta taxa de participação na plataforma, com diversidade de participação em tecnologias e muitas vitórias. Outro importante grupo é composto por usuários que estão a menos tempo na plataforma e têm dificuldade em evoluir e começar a vencer desafios. Por fim, verificou-se a existência de um grande grupo de usuários que, apesar da grande quantidade de dias necessárias para começar a competir, vence vários desafios.

Esses resultados contribuem para uma das áreas mais desafiadoras do *crowdsourcing* na atualidade que é o problema da divisão ou granularidade das tarefas (Saremi et al., 2017; Tong et al., 2018), pois foi possível verificar que existem pessoas que desistem



prematuramente, ou depois que atingem um objetivo, param de contribuir. Deste modo, tarefas curtas poderiam atrair e manter esse público. Por meio do perfil de contribuição dos participantes é possível verificar quais tipos de demandas cada plataforma é capaz de receber, visando ao melhor aproveitamento dos recursos humanos disponíveis. Por fim, a constituição do perfil de contribuição pode auxiliar na compreensão de outro ponto problemático do *crowdsourcing* retratado por Zanatta et al. (2017) e verificado no decorrer deste trabalho, que é a dificuldade encontrada pelos novos usuários em contribuir com o *crowdsourcing*, ocasionando um alto nível de desistência dos desenvolvedores.

## 1.1 Contribuições

Este trabalho analisou o perfil de contribuição dos usuários de *crowdsourcing* na plataforma Topcoder, e, conforme os resultados anteriormente citados, as contribuições são:

- **Caracterização do atributo formador de perfil de contribuição denominado hiperespecialista.** Por meio da análise da hiperespecialização foi possível verificar que existem diversos desenvolvedores detentores de muitas participações e de muitas vitórias na plataforma. Este fato independe da condição do usuário ser hiperespecialista ou não. Esta observação indica que não somente os hiperespecialistas possuem espaço na plataforma, contrariando a visão de Malone e seus colegas em 2011.
- **Evidência da grande taxa de desistência de novatos.** Na primeira etapa da fase 1, foi possível verificar que muitos usuários param de competir prematuramente. Identificou-se que 66% dos usuários analisados aparecem nos primeiros seis meses após sua inscrição na plataforma e não retornam.
- **Classificação e catalogação dos perfis de contribuição dos usuários com base em suas ações.** Por meio da busca por perfis de contribuição semelhantes foram verificados cinco grupos distintos. A verificação da existência de usuários com muitas submissões e com muitas vitórias encontrados na primeira fase é corroborado pelos grupos de usuários elite e tardios encontrados na segunda fase deste trabalho. Além disso, foi possível observar a dificuldade encontrada por alguns usuários em manter-se competindo, visto que os novatos possuem baixa taxa de dias em competição. Por fim, a principal contribuição da segunda fase é a divisão dos usuários em grupos criados com base na similaridade dos perfis de cada usuário, bem como a contextualização de cada grupo.

## 1.2 Outras contribuições

Além das contribuições citadas anteriormente, resultaram deste trabalho outros artefatos:

- Desenvolvimento de um sistema de coleta de dados da plataforma Topcoder. O sistema foi implementado em Java e está disponível<sup>3</sup>;
- Conjunto de dados do Topcoder. O banco de dados gerado pela coleta e utilizado neste trabalho está disponível neste endereço<sup>4</sup>. Com o intuito de preservar a identidade dos usuários, seus nomes (apelidos) foram removidos da base.

## 1.3 Artigos

Até o presente momento, dois artigos foram concebidos com base nos resultados deste trabalho, sendo eles:

- **Estudo exploratório sobre o perfil de contribuição dos desenvolvedores da plataforma Topcoder**, submetido para o *VII Workshop de Teses e Dissertações do CBSOft*. Este foi aceito e apresentado;
- **Uma Análise Preliminar da Hiperespecialização no Contexto de *Crowdsourcing* na Plataforma Topcoder**, submetido para o *V Workshop on Software Visualization, Evolution and Maintenance*. Este foi aceito, apresentado e recebeu o prêmio de terceiro melhor artigo da edição do *workshop*;
- ***Characterizing the hyperspecialists in the context of Crowdsourcing Software Development***, submetido para o *Journal of the Brazilian Computer Society* e se encontra em processo de revisão.

Pretende-se ainda consolidar os resultados da segunda fase deste estudo e submeter um terceiro artigo.

---

<sup>3</sup><https://bitbucket.org/anderson-neira/dissertacao/src>

<sup>4</sup>[https://bitbucket.org/anderson-neira/dissertacao/src/master/banco/banco\\_de\\_dados.sql.rar](https://bitbucket.org/anderson-neira/dissertacao/src/master/banco/banco_de_dados.sql.rar)

---

## Revisão de literatura

---

A popularização da internet proporcionou uma variedade de serviços que mudou a forma que os seres humanos vivem e trabalham (Messenger et al., 2017), seja no âmbito da comunicação instantânea, com a veloz propagação das informações, proporcionando novas formas de ganhar dinheiro, ou apenas garantindo uma nova forma de entretenimento.

Um dos pontos mais importantes da internet é a virtual aproximação de pessoas fazendo com que modelos como o *crowdsourcing* possam ser viáveis. A união de pessoas de diversas partes do mundo é um movimento muito forte, como foi retratado por Cooper et al. (2010), em que um grupo de participantes de uma plataforma chamada FoldIt<sup>1</sup> resolveu um problema relacionado à dobra de proteínas que há anos intrigava pesquisadores. Outro modo de verificar a força das pessoas nas comunidades online são as plataformas Yahoo! Answers<sup>2</sup> e Stack Overflow<sup>3</sup>, pois elas dependem das contribuições das pessoas para difundir o conhecimento.

Este capítulo aborda, na seção 2.1, definições a respeito de *crowdsourcing*, bem como características das plataformas que disponibilizam este tipo de serviço. Na seção 2.2 são descritos trabalhos relacionados ao *crowdsourcing* e características sobre o perfil de contribuição dos participantes em comunidades online.

---

<sup>1</sup>[www.fold.it](http://www.fold.it)

<sup>2</sup>[www.answers.yahoo.com](http://www.answers.yahoo.com)

<sup>3</sup>[www.stackoverflow.com](http://www.stackoverflow.com)

## 2.1 Fundamentação teórica

Para Howe (2006) *crowdsourcing* representa a ação de uma empresa designar tarefas a fim de que uma vasta rede de pessoas possa trabalhar nelas. Este modelo viabilizou a resolução de problemas que anteriormente poderiam ser praticamente insolúveis por requerer uma grande quantidade de interação humana, de pessoas com habilidades particulares, que são localmente escassas ou devido ao alto custo de contratação. Este modelo também proporcionou melhoria na resolução de problemas relativamente comuns, como transcrever o texto de uma imagem ou testar software, contudo acabou expondo necessidades que ficavam restritas às empresas.

No âmbito do desenvolvimento de software, o *crowdsourcing* também tem sido observado nos últimos anos. Mao et al. (2015a) dizem que o *crowdsourcing* para desenvolvimento de software pode ser entendido como a ação de realizar qualquer tarefa no âmbito de engenharia de software fora da empresa. Estas demandas são disponibilizadas para um grupo geralmente grande de pessoas, possibilitando que elas decidam em quais tarefas vão atuar. O *crowdsourcing* faz uso da inteligência e das contribuições de muitas pessoas com o intuito de atingir um objetivo específico, seja o desenvolvimento de software, teste de software, ou simplesmente compartilhamento de conhecimento (Stol et al., 2017). Ainda para Stol et al. (2017), o futuro da engenharia de software e do desenvolvimento de software envolverá contribuições que transcenderão aos domínios de uma empresa, e uma das formas contará com a força das comunidades online.

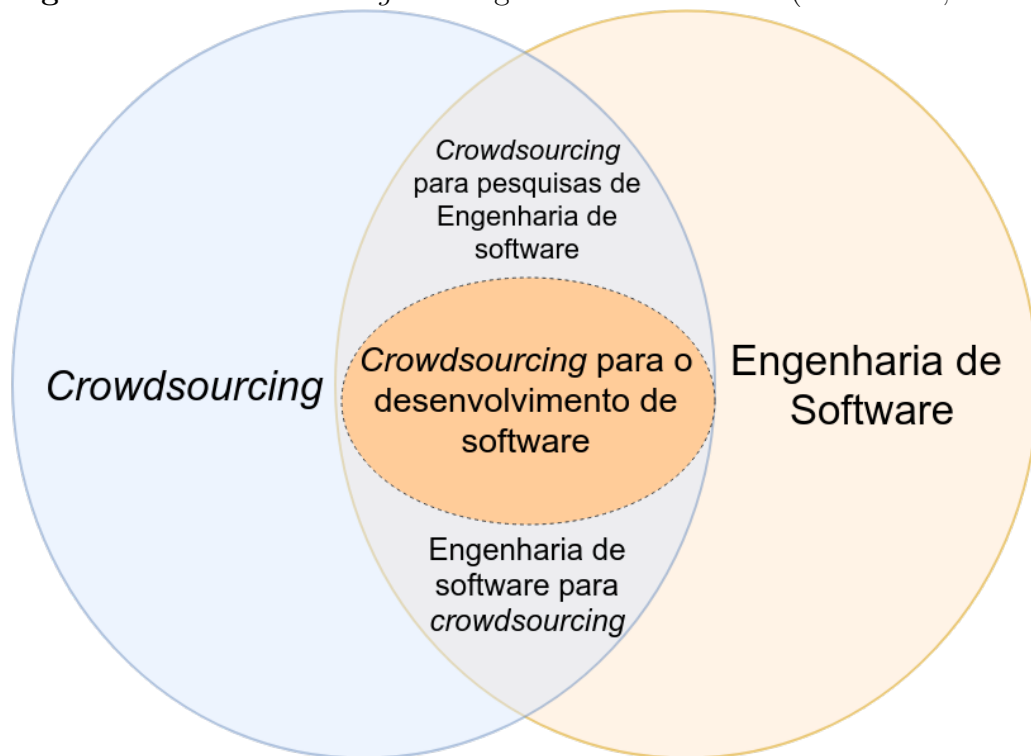
### 2.1.1 Contextualização do crowdsourcing

Ao longo dos anos, a engenharia de software vem propondo padronizações nos ciclos de desenvolvimento de software. Contudo o *crowdsourcing* está aos poucos oferecendo uma alternativa ao desenvolvido de software tradicional (LaToza e van der Hoek, 2016). O *crowdsourcing* para desenvolvimento de software pode auxiliar a engenharia de software em diversas áreas da implementação dos softwares, por exemplo, fazer o levantamento de requisitos (Alvertis et al., 2016; Breaux e Schaub, 2014), projetar interfaces (Lasecki et al., 2015), desenvolver artefatos (de Souza et al., 2014), testar software (Liu et al., 2012) entre outros.

Mao et al. (2015a) por meio da Figura - 2.1 retratam as possibilidades que a união do *crowdsourcing* e da engenharia de software oferecem:

- *Crowdsourcing* para pesquisas de Engenharia de software: qualquer pesquisa que envolva seres humanos pode utilizar técnicas de *crowdsourcing* para apoiar o recrutamento de pessoas;
- Engenharia de software para *crowdsourcing*: técnicas de engenharia de software podem ser utilizadas para apoiar a implementação de *crowdsourcing* genérico;
- *Crowdsourcing* para o desenvolvimento de software (Crowdsourced Software Engineering (CSE)): deriva do *crowdsourcing* definido por Howe (2006), pois, por meio de uma chamada aberta, recruta mão de obra para contribuir em vários tipos de tarefa de engenharia de software.

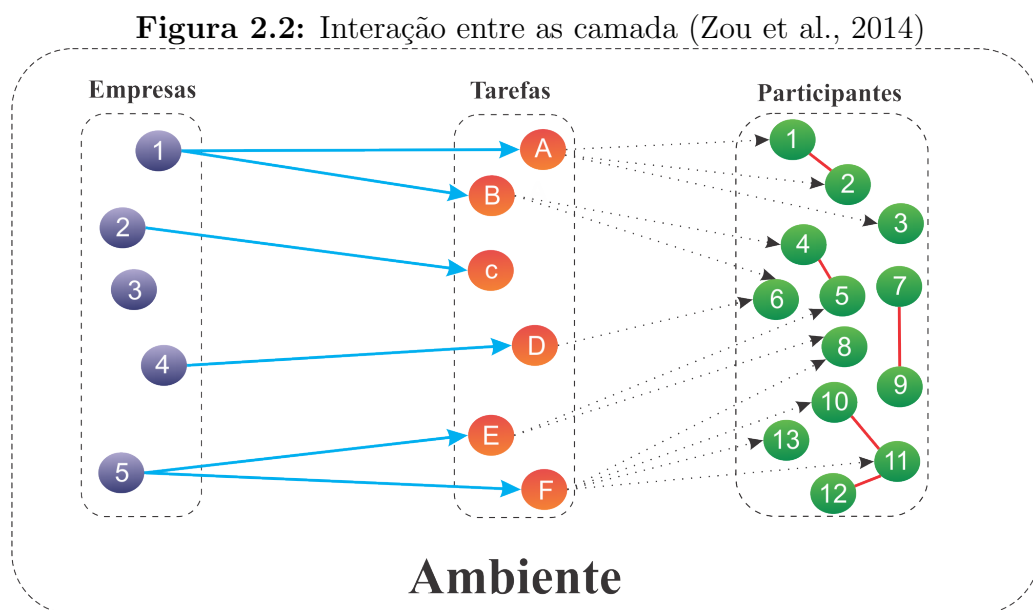
**Figura 2.1:** *Crowdsourcing* e a Engenharia de Software (Mao et al., 2015a)



### 2.1.2 Interação do modelo

Hosseini et al. (2014) defendem a existência de quatro pilares que devem ser considerados para a prosperidade do *crowdsourcing* no desenvolvimento de software: as **empresas** (clientes que precisam de software); os **trabalhadores** (usuários que desenvolvem artefatos relacionados à software); a **plataforma** (gerencia a interação entre as empresas e os trabalhadores); e as **tarefas** (definição do que deve ser feito).

Por meio da Figura - 2.2, Zou et al. (2014) retratam como acontece a interação entre os pilares do *crowdsourcing*. Inicialmente, a plataforma provê o ambiente em que as empresas e os participantes estarão imersos. Em geral, é de responsabilidade da plataforma atrair, manter e dar suporte para que as empresas possam extrair o máximo da comunidade. Algumas plataformas também auxiliam na capacitação dos usuários. As empresas solicitam as tarefas segundo suas necessidades. A tarefa em si é o conjunto de especificações, materiais de apoio e datas limites sobre cada uma das necessidades solicitadas pelos clientes. É possível e recomendado que existam mais de uma tarefa da mesma empresa ou do mesmo projeto disponível ao mesmo tempo (tarefas paralelizadas). Em algumas plataformas, é possível contar com a colaboração de membros qualificados da plataforma durante todo o processo de gerência dos projetos que contemplam desde a criação das tarefas, suporte aos membros da comunidade até a junção dos artefatos em uma solução final. Por fim, os participantes ou desenvolvedores escolhem em quais tarefas vão competir. Um aspecto importante é a possibilidade de interação entre participantes através de fóruns, para que possam sanar eventuais dúvidas, compartilhar experiências e assim concluir as tarefas da melhor forma possível.



### 2.1.3 Vantagens e desvantagens

Para atrair e manter as pessoas ativas nas comunidades de *crowdsourcing* é necessário existir vantagens para quem faz as contribuições. Uma das principais vantagens que atrai usuários é a recompensa ofertada pela resolução das tarefas. Na maior parte dos casos, o

criador da melhor contribuição é quem recebe o valor financeiro ofertado pela resolução da tarefa (LaToza e van der Hoek, 2016). Também existem casos que, além do dono da melhor contribuição, os usuários que ficaram bem classificados, por exemplo, o segundo e terceiro colocados, também recebem uma recompensa em dinheiro, porém o valor é, em geral, menor. Uma variação deste modelo pode ser a busca por inconsistências nos softwares. Esse tipo de tarefa pode recompensar todos os competidores que encontrarem qualquer erro, ou ainda recompensar o competidor que reportar mais defeitos. A média dos valores pagos varia entre as plataformas, e o método de pagamento pode variar também. Outra característica vantajosa para quem contribui é a oportunidade de trabalhar em tarefas de empresas de reconhecimento mundial (Saremi e Yang, 2015), como a IBM<sup>4</sup>, NASA<sup>5</sup>, eBay<sup>6</sup> e Google<sup>7</sup> entre outras. Ganhar competições de empresas de renome internacional é de extrema importância, pois, além do reconhecimento proporcionado pela da comunidade, essas participações podem se transformar em oportunidades profissionais. Por fim, o usuário que contribui com o *crowdsourcing* aumenta sua experiência e aprimora seus conhecimentos, pois está lidando com projetos reais de empresas de todo o mundo.

O *crowdsourcing* também proporciona vantagens para a empresa. O *time-to-market* é um fator decisivo para o sucesso de uma ideia. Então, economizar tempo é importante. O *crowdsourcing* torna isso possível, pois as empresas podem alocar várias tarefas para serem desenvolvidas em paralelo, fato esse que se torna limitado com uma equipe nas *softwares houses* (LaToza e van der Hoek, 2016). Como muitas pessoas são atraídas pelas tarefas, e existe grande diversidade de conhecimentos e experiências, a chance de encontrar membros capacitados para resolvê-las é alta, fato esse que proporciona a qualidade esperada dos artefatos entregues. Além de economizar tempo sem afetar a qualidade dos projetos, ainda é possível reduzir seus custos, pois geralmente não há a necessidade de contratar, diretamente, pessoas para a equipe, ou investir em estrutura para mantê-los produzindo. Deste modo, é possível economizar de 30% a 80% do valor que seria necessário para desenvolver um software por meio dos métodos tradicionais (Hasteer et al., 2016; Lydon, 2012).

Contudo existem alguns pontos que podem influenciar as empresas a não adotarem o *crowdsourcing*. Alguns projetos podem ser sigilosos, pois nenhuma informação pode vir a público antes do lançamento do produto. Este fato inviabiliza a utilização de plataformas de *crowdsourcing*, pois os detalhes dos projetos ficam disponíveis para que qualquer pessoa possa se candidatar a fazer a tarefa. Outro problema que pode afetar a adoção do

---

<sup>4</sup>[www.ibm.com](http://www.ibm.com)

<sup>5</sup>[www.nasa.gov](http://www.nasa.gov)

<sup>6</sup>[www.ebay.com](http://www.ebay.com)

<sup>7</sup>[www.google.com.br](http://www.google.com.br)

*crowdsourcing* são algumas contribuições com conteúdo inapropriado. Um exemplo disso é apresentado por Dwarakanath et al. (2016) na Figura - 2.3, em que é retratada uma tarefa cujo objetivo é a concepção de uma nova embalagem para um produto. Contudo algumas das soluções enviadas eram brincadeiras feitas na internet, fato esse que aumenta os esforços para selecionar os trabalhos sérios e escolher o vencedor. Especificamente no âmbito de *crowdsourcing* para o desenvolvimento de software, existem ainda outras preocupações. Uma delas é a possível dificuldade de manter e evoluir as partes do software que vieram do *crowdsourcing*, pois o código pode não seguir os padrões da empresa, ou ainda usar recursos que não são familiares à equipe. Por fim, outro problema que as empresas podem encontrar é a má intenção de algumas pessoas. Elas podem colocar, propositalmente, códigos maliciosos, ou ainda, deixar falhas para que eles possam agir de má-fé, podendo causar prejuízos.

**Figura 2.3:** Soluções enviadas por usuários do *crowdsourcing* (Dwarakanath et al., 2016)



#### 2.1.4 Plataformas de crowdsourcing

Para LaToza e van der Hoek (2016), as plataformas que implementam o *crowdsourcing* podem operar de três maneiras distintas, sendo elas: *peer production*; modelo de competições; e o modelo de micro-tarefas:

- *Peer production*: os participantes colaboram juntos para a construção de um único artefato e, em geral, não há pagamento por essas colaborações. Os usuários são motivados, majoritariamente, pela oportunidade de adquirir novas habilidades, aprimorar conhecimentos. Este paradigma assemelha-se ao adotado pelo software livre, por exemplo;
- Modelo de competições: as empresas descrevem suas necessidades em forma de tarefas e abre concorrência pública pela melhor contribuição, que é paga em dinheiro



ao desenvolvedor. A plataforma Topcoder foi a pioneira neste paradigma no *crowdsourcing* para o desenvolvimento de software;

- Modelo de micro-tarefas: as necessidades da empresa-cliente são decompostas em pequenas tarefas que podem ser concluídas rapidamente e são disponibilizadas para a comunidade. O dono da melhor contribuição recebe a recompensa ofertada pelo trabalho, que posteriormente, será anexado aos resultados das outras tarefas, completando a da demanda empresa.

Existem diversas plataformas que utilizam o modelo de *crowdsourcing*. Uma das maiores e mais influentes que seguem o padrão de micro-tarefas é a *Amazon Mechanical Turk* - AMT<sup>8</sup> (LaToza e van der Hoek, 2016; Stol et al., 2017). Com a AMT é possível encontrar pequenas tarefas sobre diversos temas como, por exemplo, tradução de pequenos textos entre várias línguas; ouvir uma gravação e transcrever o conteúdo; responder pesquisas; coletar dados de sites; entre outros assuntos possíveis. Uma característica do AMT é que os cadastros são auditados e, dependendo da disponibilidade de tarefas para a localidade o cadastro, pode ser negado, justamente porque as tarefas podem ser disponibilizadas apenas para um país, por exemplo.

No contexto de *crowdsourcing* para desenvolvimento de software existem algumas plataformas disponíveis no mercado, como por exemplo, a plataforma 99designs<sup>9</sup>, que utiliza o modo de operação baseado em competições para criar projetos relacionados ao desenvolvimento de layouts de sites, logos, ícones entre outros serviços disponibilizados (LaToza e van der Hoek, 2016). A busca por erros em sistemas, *bug bounties*, é uma área popular em *crowdsourcing* (LaToza e van der Hoek, 2016), que também utiliza o modelo de competições para as multidões testarem sistemas de software. Sobre essa prática, é oportuno destacar as plataformas Pay4Bugs<sup>10</sup> e a Bugcrowd<sup>11</sup>. A plataforma Upwork<sup>12</sup> adota um modelo de competição semelhante ao citado anteriormente para o desenvolvimento de software em geral. Com ela é possível solicitar demandas no formato de tarefas, que são disponibilizadas para os membros da plataforma. Cada participante decide em quais tarefas vai trabalhar, contudo o dono da tarefa escolhe os participantes que ele julga estarem mais aptos e faz uma entrevista com eles. Após as entrevista, o escolhido é contratado para completar a tarefa (Abhinav et al., 2016; Dubey et al., 2016).

---

<sup>8</sup>[www.mturk.com](http://www.mturk.com)

<sup>9</sup>[www.99designs.com.br](http://www.99designs.com.br)

<sup>10</sup>[www.pay4bugs.com](http://www.pay4bugs.com)

<sup>11</sup>[www.bugcrowd.com](http://www.bugcrowd.com)

<sup>12</sup>[www.upwork.com](http://www.upwork.com)

Mao et al. (2015a) retratam, por meio da Tabela - 2.1, algumas plataformas cujo objetivo é auxiliar no processo de desenvolvimento de software. Nela é possível observar o nome, o ano de criação da plataforma, o domínio de atuação e o formato de chamada. No caso do domínio, algumas plataformas são especializadas em partes do desenvolvimento de software (como teste aplicativos para dispositivos móveis e teste de segurança); outras abrangem todo o processo de desenvolvimento de software. Também existem modos diferentes de escolher os usuários que vão competir. Existem casos que todos os inscritos podem participar (competição online) e casos que os usuários solicitam a participação, e o dono da tarefa escolhe quem vai competir (correspondência sob demanda).

**Tabela 2.1:** Plataformas de *crowdsourcing* para o desenvolvimento de software (Mao et al., 2015a)

Plataforma	Criação	Domínio	Formato de chamada
Topcoder	2001	Desenvolvimento de software	Competição online
Bountify	2013	Pequenas tarefas de codificação	Competição online
Bugcrowd	2012	Teste de segurança de software	Competição online
uTest	2007	Teste de software	Competição online
Passbrains	2012	Teste de software	Correspondência sob demanda
TestBirds	2011	Teste de software	Correspondência sob demanda
99Tests	2010	Teste de software	Correspondência sob demanda
Mob4hire	2008	Teste de aplicativos móveis	Correspondência sob demanda

### 2.1.5 Topcoder

A plataforma escolhida para ser utilizada neste trabalho é a Topcoder. Ela atrai empresas com necessidades diferentes, como design telas, criação de logos, testes de software, desenvolvimento de software, otimização de funções, entre outras demandas que podem ser solicitadas. Empresas de fama internacional utilizam este serviço, por exemplo, NASA<sup>13</sup>, IBM<sup>14 15</sup>, eBay<sup>16</sup>, Honeywell<sup>17</sup>. A própria plataforma solicita a seus membros a conclusão de tarefas para criar funcionalidades ou melhorar a plataforma<sup>18</sup>. Semelhante ao processo da Upwork, a Topcoder disponibiliza as tarefas de seus contratantes para a comunidade trabalhar, competindo nas demandas disponíveis. Geralmente muitos membros se inscrevem para completar as tarefas na Topcoder, e várias delas possuem

<sup>13</sup><http://crowdsourcing.topcoder.com/project/nasa-iss-fit-ipad-app/>

<sup>14</sup><http://crowdsourcing.topcoder.com/project/ibm-ifundit/>

<sup>15</sup><http://crowdsourcing.topcoder.com/project/ibm-media-library/>

<sup>16</sup><http://crowdsourcing.topcoder.com/project/ebay-gethelp-mobile-app/>

<sup>17</sup><http://crowdsourcing.topcoder.com/project/honeywell/>

<sup>18</sup><https://www.topcoder.com/challenge-details/30056780/?type=develop>

muitas contribuições (Dubey et al., 2016). A submissão é a ação do usuário enviar a sua resolução da tarefa, artefato de software, para a tarefa. Posteriormente a equipe criadora da tarefa é responsável por avaliar todas as submissões, escolher e pagar a recompensa para o desenvolvedor dono da melhor submissão. Em alguns casos, paga-se também aos criadores das submissões que ficaram em segundo e terceiro lugares, mas esse é um fato que varia entre as tarefas e é decidido pela empresa contratante.

A Topcoder classifica as demandas dos clientes em três categorias diferentes: desenvolvimento (do inglês *development*), *design* e Ciência de Dados (do inglês *data science*). A categoria desenvolvimento recebe tarefas relacionadas à programação de softwares, teste de softwares, melhorias de códigos entre outros. Existem algumas subdivisões relacionadas aos desafios de desenvolvimento<sup>19</sup>, sendo elas *Code*, *Assembly*, *First2Finish*, *UI Prototype* e *Architecture*. O *design* relaciona as tarefas cujo intuito é construir artefatos relativos a telas, logos ou imagens. Por fim, a categoria Ciência de Dados agrupa tarefas reais bem como tarefas que são criadas para desafiar os membros da comunidade. A categoria Ciência de Dados recebe desafios que envolvem grande quantidade de dados. Além desses desafios, existem outras duas subcategorias que pertencem a Ciência de Dados, as *marathons* e as *Single Round Matches* (SRM). As *marathons* são competições focadas na resolução de problemas algorítmicos mais difíceis, que podem ser hipotéticos ou reais. As SRMs são desafios que os membros mais qualificados da plataforma criam para que os outros possam resolver. Esse tipo de competição é dividida em dois grupos, os mais experientes e os menos experientes, baseados na pontuação alcançada na plataforma. A cada rodada, geralmente realizada a cada quinze dias, os participantes devem resolver três desafios de diferentes níveis. Após o término do tempo, eles competem com o objetivo de propor falhas nos algoritmos dos outros competidores. Todas as ações são pontuadas, e, após o final de cada desafio, o maior pontuador de cada divisão é eleito vencedor.

Mao et al. (2015b) definem, por meio da Figura - 2.4, como o processo de desenvolvimento de software utilizado pela Topcoder mantém relação com as fases de cada tarefa. No lado esquerdo (a) da figura estão descritas todas as fases do desenvolvimento de software suportado pela plataforma. Inicialmente, na fase dos requisitos são definidos os objetivos dos projetos bem como todo o planejamento e orçamento estimado para a conclusão do mesmo. Na fase de arquitetura, o projeto é dividido em componentes, pois a plataforma adota a prática de desenvolvimento baseado em componentes. Ao término da fase de arquitetura, é esperado que todos os componentes possuam a documentação necessária para o desenvolvimento. A documentação geralmente é composta por diagramas da *Unified Modeling Language* (UML) e as especificações dos componentes. Os componentes

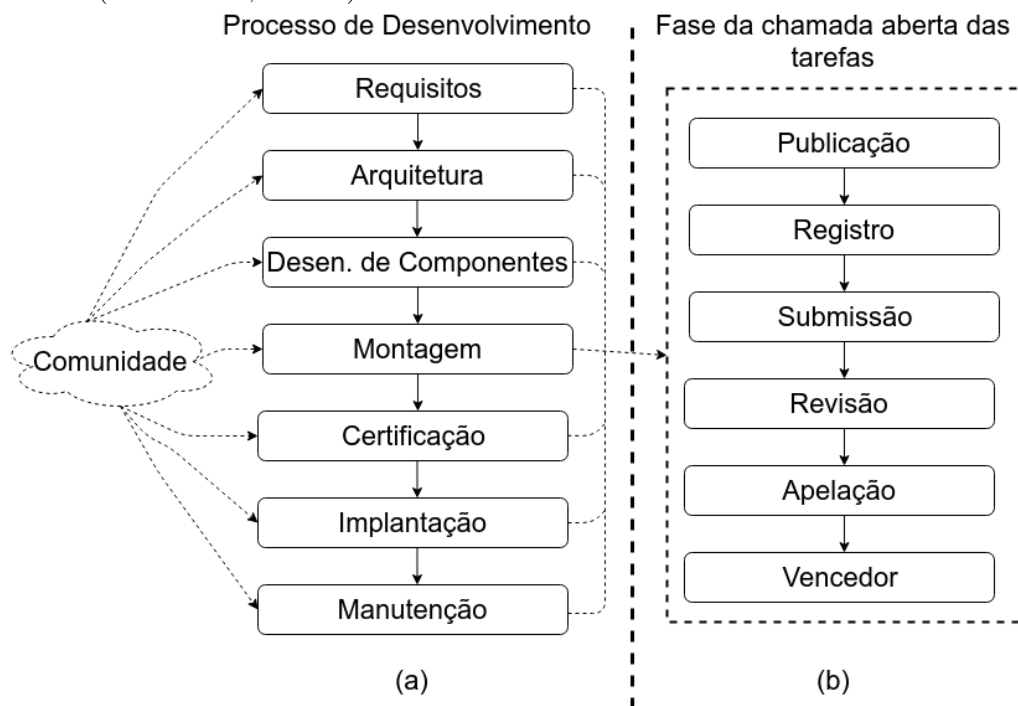
---

<sup>19</sup><https://www.topcoder.com/member-onboarding/competing-in-development-challenges/>

especificados são criados na fase de desenvolvimento e, após todos os componentes terem sido desenvolvidos, os mesmos são combinados na fase de montagem. O projeto é enviado para ser validado pela equipe de qualidade do cliente. Para auxiliar o cliente a testar e manter o projeto, a fase de manutenção fornece modos para a comunidade encontrar e reportar erros na aplicação.

Todas as fases especificadas do lado esquerdo (a) da Figura - 2.4 podem contar com a participação da comunidade, seguindo as etapas do lado direito (b) da Figura - 2.4. Inicialmente, a tarefa é publicada contendo a explicação do que deve ser feito e dois prazos importantes são definidos: a data do limite da inscrição na tarefa e a data limite da entrega do artefato de software. Após a publicação, os usuários que desejarem participar devem se inscrever até o prazo estabelecido e, posteriormente, devem submeter sua contribuição até o prazo estabelecido. Tendo a data limite de submissão encerrada, as soluções são avaliadas por revisão feita em pares. Após essa revisão, as soluções aprovadas recebem uma nota, segundo a visão de três outros avaliadores. Após a fase de revisão, os donos das soluções são avisados sobre as classificações, e, caso eles não concordem com a revisão, têm o direito de argumentar durante a fase de apelação. Depois de concluída a fase de apelação, os vencedores são anunciados e recebem a recompensa conforme a colocação atingida.

**Figura 2.4:** Processo de desenvolvimento de software utilizado pela plataforma Topcoder (Mao et al., 2015b)



Todas as tarefas que estão abertas para registros podem ser encontradas neste endereço eletrônico<sup>20</sup>. Um exemplo de tarefa pode ser encontrado neste outro endereço<sup>21</sup>. Neste exemplo, é solicitado que os participantes façam algumas alterações no site como adicionar uma geração de chave no login dos usuários, adicionar uma animação na tela inicial, arrumar o menu para visualização *mobile* entre outras coisas solicitadas nesta tarefa. O dono da melhor contribuição recebe mil e duzentos dólares e o dono da segunda melhor contribuição recebe seiscentos dólares.

## 2.2 Revisão bibliográfica

Diversos são os estudos conduzidos na área de *crowdsourcing* abrangendo temas como recomendação de participantes, motivação dos membros da comunidade, interferência da recompensa na construção de contribuições e o perfil de contribuição dos participantes. Nesta seção serão apresentados alguns desses trabalhos e como eles se relacionam ao trabalho aqui proposto.

### 2.2.1 Recomendação de participantes

O modelo de *crowdsourcing* utiliza a diversidade de experiências e conhecimentos dos membros da comunidade para atrair as empresas que investem tempo e dinheiro, disponibilizando tarefas a serem completadas pelos participantes.

Um dos maiores problemas de modelos que usam a estratégia de “chamada aberta” é selecionar as pessoas apropriadas para fazer cada tarefa. (Tunio et al., 2017). Desse modo, é importante que as pessoas mais qualificadas possam encontrar e completar os desafios, justamente porque o resultado tende a ser uma contribuição que satisfaça a expectativa da empresa (Ye e Wang, 2016; Yuen et al., 2012). Com o objetivo de prover uma alta qualidade das submissões dos participantes, alguns autores dispuseram seus esforços para criar abordagens capazes de sugerir as pessoas mais aptas para as tarefas disponíveis. Alguns desses estudos focam em recomendar participantes nas plataformas cujo objetivo é a resolução de tarefas em geral (Ho e Vaughan, 2012; Qiu et al., 2016; Ye e Wang, 2016; Yuen et al., 2012), mas essa também é uma preocupação do *crowdsourcing* que disponibiliza tarefas relacionadas ao desenvolvimento de sistemas (Karim et al., 2016; Mao et al., 2016, 2015b).

---

<sup>20</sup><https://www.topcoder.com/challenges>

<sup>21</sup><https://www.topcoder.com/challenges/30067487>

Independente do ambiente e do domínio, a abordagem de criação do modelo é semelhante. Em geral, utiliza-se uma base com dados sobre os membros da plataforma e as tarefas que eles desempenharam. Informações tais como valor da recompensa, tecnologias, título, descrição e as datas de criação e de encerramento são utilizadas para construir o perfil de participação dos membros. A partir disso o modelo começa a sugerir quais são os participantes que mais se adequam ao perfil da tarefa.

Apesar da importância de prever as pessoas mais aptas para uma tarefa Karim et al. (2016), também focaram em outros aspectos importantes. Eles buscaram um modo de identificar as pessoas que não seriam as vencedoras do desafio. Desta maneira é possível economizar o tempo gasto com uma tarefa que poderia não ser a mais adequada, e ainda diminuir a concorrência, fazendo com que o participante fique disponível para disputar uma tarefa condizente ao seu perfil.

Algumas abordagens obtiveram resultados melhores quando comparadas a outras. Por exemplo, o trabalho de (Karim et al., 2016) conseguiu uma taxa de acerto de cerca de 94%, quando o objetivo era prever um participante que ficaria entre os dois mais bem colocados na tarefa. Outro resultado decorrente da pesquisa de (Karim et al., 2016) é que com a recomendação de tarefas em um período de 30 dias foi possível economizar cerca de 3,5 dias de membros mais experientes e cerca de 4.6 dias para membros menos experientes.

## 2.2.2 Motivação

Além de contar com a “sabedoria das multidões”, existem alguns fatores que influenciam diretamente na qualidade das atividades realizadas por meio de modelo *crowdsourcing*. Um deles é a motivação que os participantes possuem para poder contribuir nas plataformas. É de suma importância que os participantes estejam motivados para que o resultado seja compatível com as necessidades das empresas.

Alguns exemplos de fatores que podem atrair e motivar os participantes das plataformas de *crowdsourcing* são a recompensa oferecida para os melhores projetos e a reputação das empresas que solicitam as tarefas. De acordo com Kaufmann et al. (2011), Rogstadius et al. (2011) e Hossain (2012), existem dois grupos de fatores que motivam os participantes de *crowdsourcing*: o grupo dos fatores extrínsecos e o dos intrínsecos. Os fatores extrínsecos levam em consideração as ações relacionadas à recompensa que são obtidas por meio das competições. Essas recompensas podem ser o dinheiro disponibilizado para quem cumprir a tarefa e ainda, no caso de desenvolvimento de software, a recompensa pode ser o conhecimento adquirido ou até mesmo notoriedade na indústria, podendo proporcionar

oportunidades no mercado de trabalho. Em geral os fatores intrínsecos são mais difíceis de serem alcançados, pois, ao contrário dos extrínsecos, nesses fatores não é esperado recompensa ou outro ganho material para que o participante seja motivado. Geralmente as pessoas motivadas com fatores dessa classificação participam dos desafios em busca da satisfação pessoal. Fatores intrínsecos possuem uma influência positiva sobre a qualidade dos resultados, pois os participantes tendem a competir com mais criatividade quando não estão pressionados pela necessidade de ganhar dinheiro, por exemplo, sendo, portanto os fatores intrínsecos, em alguns casos, mais importantes que os extrínsecos (Pinto e dos Santos, 2018).

### 2.2.3 Recompensas e contribuições

Dentre vários fatores de motivação, um dos que mais influenciam na assiduidade dos participantes é a recompensa em dinheiro oferecida pela resolução das tarefas. Esse fato pode ser observado no *crowdsourcing* em geral. Segundo dados oficiais do site do Topcoder<sup>22</sup>, foram distribuídos mais de oitenta milhões de dólares em recompensa desde sua fundação em 2001, tendo um usuário recebido, sozinho, mais de seiscentos mil dólares em recompensas<sup>23</sup>.

Dada a importância das recompensas, alguns autores buscam entender a relação destas com o comportamento dos participantes de *crowdsourcing*. Dwarakanath et al. (2016) citam que, na plataforma AMT, as tarefas que possuem uma recompensa mais alta tendem a atrair participantes que não são confiáveis, ou seja, pessoas que não se comprometem em completar o solicitado com qualidade, ou que, de alguma maneira, podem atrapalhar o pleno cumprimento das tarefas. Em geral, estes buscam apenas a recompensa.

Para Yang e Saremi (2015) a recompensa deve ser o suficiente para atrair os trabalhadores, contudo, conforme as recompensas pagas para os vencedores aumentam, ocorre uma diminuição na quantidade e na qualidade da média das submissões, isto devido ao fato de o número de registrados também diminuir. Um dos fatores que motivam este comportamento é a natural associação de que as tarefas complexas possuem recompensas maiores.

Para possibilitar o melhor aproveitamento dos recursos, Yang e Saremi (2015) encorajam dividir a demanda em partes menores, buscando atingir um equilíbrio no custo-benefício da tarefa, a fim de possibilitar que os especialistas possam interessar-se e completar as tarefas com boas soluções (LaToza e van der Hoek, 2016). Contudo

---

<sup>22</sup><https://www.topcoder.com/marketplace/the-community/>

<sup>23</sup>[https://www.topcoder.com/tc?module=ComponentRecordbook&c=overall\\_money](https://www.topcoder.com/tc?module=ComponentRecordbook&c=overall_money)

atingir um custo-benefício adequado para a tarefa é um grande desafio, pois, segundo Karim et al. (2016), cerca de 82.9% dos inscritos nas tarefas, da plataforma Topcoder, acabam desistindo de completá-las, e cerca de 55.8% das submissões feitas não passam pelo processo de revisão, ou seja, devido à baixa qualidade essas submissões não chegam a concorrer à recompensa ofertada.

## 2.2.4 Perfil de contribuição dos participantes

Entender o perfil de contribuição dos participantes de comunidades online não é uma preocupação nova e exclusiva do *crowdsourcing*. Anteriormente, houve esforços a fim de entender traços dos perfis de contribuição das pessoas que interagem com a Wikipédia<sup>24</sup>. Por exemplo, Zhang et al. (2012) estudou quanto tempo os editores da Wikipédia se mantinham contribuindo com a plataforma. O resultado foi que o tempo médio de contribuição é de 53 dias. Foi identificado também que quanto mais tempo as pessoas passam editando artigos, maior é a quantidade de dias contribuindo com a plataforma. Outra característica a respeito do perfil dos contribuintes da Wikipédia foi esclarecida pelo trabalho de Kittur et al. (2007), verificando que no início da plataforma uma pequena quantidade de pessoas faziam a maior parte das contribuições. Esse comportamento vem sendo modificado ao longo dos anos, quando a maioria das contribuições passaram ser realizadas por muitas pessoas.

Nas plataformas de perguntas e respostas, *Questions & Answers* (Q&A) também podem ser encontrados trabalhos que buscam entender o perfil de seus contribuidores. A pesquisa de Furtado et al. (2013) possui um objetivo similar ao deste trabalho, utilizando o contexto de Q&A e várias plataformas. Nela os pesquisadores buscaram entender como os participantes produzem o conhecimento. Para isso foi examinado o perfil desses contribuidores. Um dos principais resultados foi a classificação dos contribuidores das plataformas, sendo possível propor uma caracterização de perfis de usuários existentes. Outro importante resultado foi a constatação de que o perfil muda ao longo da participação dos membros na plataforma.

Conhecer o perfil de contribuição das pessoas é um fator importante na tomada de decisões, e com o *crowdsourcing* não é diferente. Dessa maneira, existem trabalhos que retratam parte do perfil dos participantes nas plataformas de *crowdsourcing*. Por exemplo, Saremi e Yang (2015) citam que membros mais experientes da plataforma Topcoder são propensos a buscar tarefas de empresas com fama internacional, ou ainda com altas recompensas. Eles ressaltam também que as pessoas experientes produzem

---

<sup>24</sup>[www.wikipedia.org](http://www.wikipedia.org)



mais, proporcionando chances de mais vitórias em suas disputas. Mao et al. (2016) relatam que os membros mais qualificados da plataforma Topcoder enfrentarem grande concorrência, estes acabam se registrando logo que a tarefa é disponibilizada, fato esse que acaba de certa forma inibindo o cadastro de outros competidores de alto nível. Saremi et al. (2017) dizem que usuários da plataforma Topcoder que começaram a participar da plataforma depois de 2010 são mais experientes, motivados, interessados e habilidosos, mas não, necessariamente, contribuem com qualidade e velocidade. Verificaram também que os usuários que possuem classificações mais altas na plataforma podem ser mais confiáveis (usuários que completam suas tarefas com velocidade e qualidade).

Saremi e Yang (2015) e Gray et al. (2016) entendem que o contato entre as pessoas que gerenciam as tarefas e os trabalhadores influenciam benéficamente a qualidade dos artefatos entregues. Uma das possíveis razões para esse fenômeno é que, com este contato, é possível sanar dúvidas que poderiam impactar negativamente a solução submetida. Gray et al. (2016) citam, também, a existência de casos que os próprios participantes se ajudam, seja na motivação para continuar contribuindo com a plataforma, seja com a indicação de tarefas, com informações sobre o dono da tarefa ou até mesmo auxiliando a conclusão do trabalho.

Algumas pesquisas buscam compreender as ações de pessoas com má intenção, ou que não conseguem boas contribuições que nas plataformas de *crowdsourcing* em geral (Mok et al., 2016, 2015; Xu et al., 2015). A pesquisa de Gadiraju (2015) sugere classificações para esses tipos de membros cadastrados na plataforma CrowdFlower<sup>25</sup>:

- Trabalhadores inelegíveis: são as pessoas que não possuem o conhecimento necessário para terminar a tarefa;
- Enganadores rápidos: possuem um zelo em ganhar dinheiro rápido. Geralmente eles não se importam com a qualidade do seu trabalho, apenas entregam qualquer coisa a fim de receber a recompensa;
- Enganadores inteligentes: são caracterizados por terem as competências necessárias para completar a tarefa com qualidade, porém eles mascaram seu real objetivo, apesar de eles não quebrarem nenhuma regra explicitamente.
- Quebrador de regras: é uma das características mais comuns, pois as pessoas que se enquadram neste tipo, geralmente, não se importam com a veracidade das informações ou com a qualidade delas, cujos trabalhos feitos por eles são descartados, por não serem úteis para o administrador da tarefa;

---

<sup>25</sup>[www.crowdfLOWER.com](http://www.crowdfLOWER.com)

- Presas ao padrão de ouro: são as pessoas que têm as competências para as tarefas e que, geralmente, produzem respostas válidas, apesar de elas não demonstrarem comportamento malicioso, falham em resolver suas tarefas por falta de atenção, fadiga ou tédio.

## 2.2.5 Novos usuários no crowdsourcing

O fenômeno da chegada de novos usuários de *crowdsourcing* foi retratado por Zanatta et al. (2017). Os autores argumentam que não é fácil atrair e manter novos usuários e que eles enfrentam problemas similares aos enfrentados pelos novos desenvolvedores de OSS (*Open Source Software*) como, por exemplo, falta de documentação, gerenciamento inadequado de tarefas, fraca usabilidade de plataforma e problemas relacionados ao idioma. Para auxiliar os recém-chegados vencer as dificuldades, os autores recomendam as plataformas de *crowdsourcing* que invistam na formatação da documentação das tarefas, pois a informação delas é o ponto chave do sucesso do *crowdsourcing*, porém quantidade demasiada de informação prejudica os novos desenvolvedores. Além disso, seria interessante a existência de formas simples de identificar a complexidade das tarefas, com o intuito de atrair os novos usuários para contribuir inicialmente em desafios mais simples a fim de que eles possam adquirir experiência e confiança.

Machado et al. (2017) fizeram um estudo inserindo no *crowdsourcing* desenvolvedores graduados e com experiência em programação que não conheciam o modelo. O intuito era usar a visão dos programadores para entender melhor aspectos relacionados ao modelo do *crowdsourcing*, especificamente na plataforma Topcoder. Nesta primeira participação, os entrevistados puderam ressaltar algumas das principais características do modelo como, criar oportunidades para os desenvolvedores ganharem dinheiro extra, diversidades de tarefas, o fato de o próprio desenvolvedor escolher em qual desafio iria contribuir e a possibilidade de existir diversas soluções para o mesmo problema. Além disso, alguns desenvolvedores utilizaram os fóruns como um meio de interagir com a comunidade, buscando compreender melhor a tarefa. Outros desenvolvedores preferiram não utilizar os fóruns para completar as tarefas. Indo ao encontro com a visão de Zanatta et al. (2017) sobre a usabilidade de plataforma, alguns desenvolvedores citaram problemas em encontrar os materiais disponíveis para completar as tarefas, ou o material disponível não era o suficiente para o devido entendimento da tarefa. Outra dificuldade foi relacionada a tarefa, pois, para os entrevistados, foi difícil encontrar tarefas de acordo com o perfil deles.

Com o intuito de atrair, manter, aprimorar as habilidades dos usuários, a plataforma Topcoder se preocupa em criar formas para ajudar os usuários a aprimorar seus conhecimentos. Em geral, o conteúdo dessas ajudas buscam capacitar usuários para contribuir em áreas que existam abundantes ofertas de tarefas, mas a mão de obra capacitada é escassa. Uma recente ação foi a adição da linguagem Swift aos desafios da categoria *Fun Challenges*, que servem justamente para o aprimoramento de habilidades.

## 2.2.6 Considerações finais

A literatura de *crowdsourcing* de outros domínios analisa a motivação dos usuários em competir (Hosseini et al., 2014; Pinto e dos Santos, 2018), a recomendação de participantes aptos para completar as tarefas (Ye e Wang, 2016; Yuen et al., 2012) e como a recompensa ofertada atrai os usuários (Dwarakanath et al., 2016). A área de *crowdsourcing* para engenharia de software ainda está evoluindo, e pode-se notar que existem estudos relacionados à recomendação de tarefas baseados no comportamento histórico (Karim et al., 2016; Mao et al., 2015b), influência das recompensas nas contribuições (LaToza e van der Hoek, 2016; Yang e Saremi, 2015) e partes do perfil de contribuição dos usuários (Mao et al., 2016; Saremi et al., 2017). Apesar de esforços existirem em outros domínios e de algumas pesquisas identificarem traços do perfil de contribuição dos participantes do Topcoder, este trabalho estende a literatura ao explorar os perfis de contribuição dos usuários de *crowdsourcing* para desenvolvimento de software. As análises conduzidas neste trabalho buscaram identificar diferentes tipos de perfis de contribuição existentes na plataforma e não apenas traços comportamentais. A fim de garantir variados perfis serão utilizados nove atributos. Esses atributos estão descritos no capítulo 5. Além disso, o que se busca é fazer um estudo exploratório em larga escala, fazendo uso de uma base de dados com cerca de 18.000 desafios.

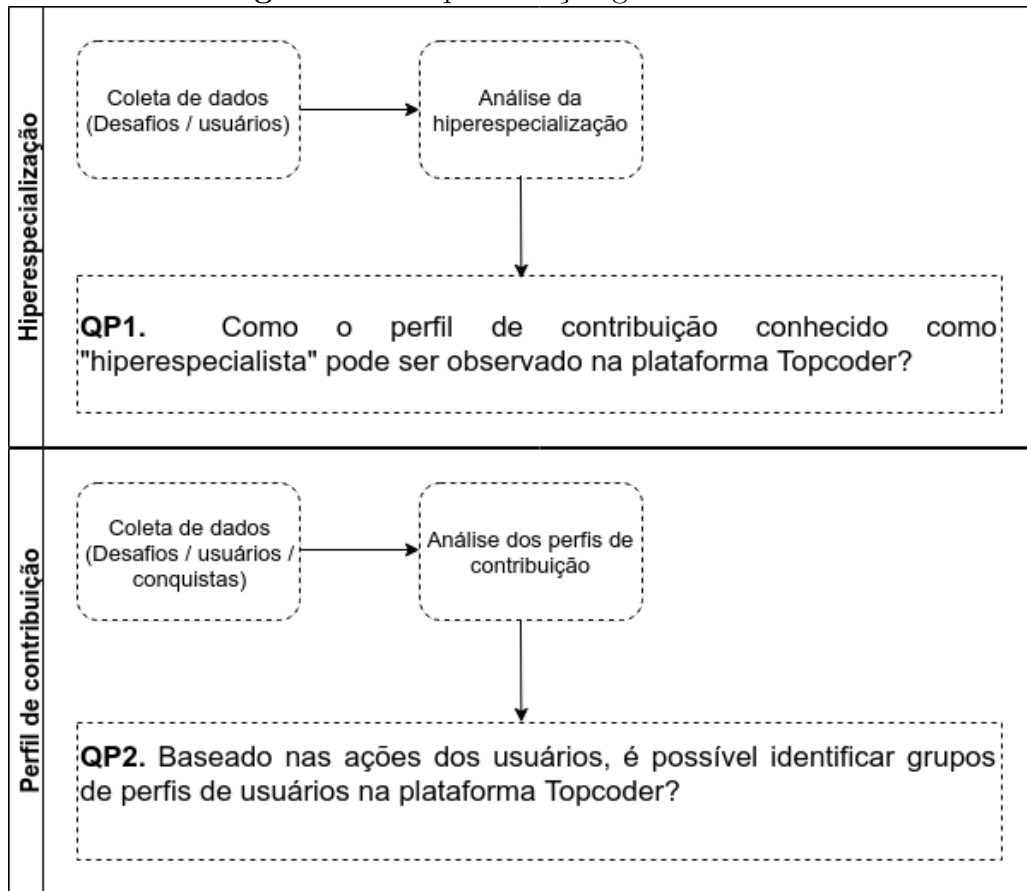
---

## Metodologia de pesquisa

---

O método deste trabalho está pautado em duas fases distintas. A Figura - 3.1 representa, em alto nível, as duas fases executadas: (i) análise da hiperespecialização e (ii) análise do perfil de contribuição. As etapas estão descritas a seguir. Essa divisão foi proposta devido ao fato de a segunda fase deste estudo utilizar a característica “hiperespecialista” para formar os grupos de usuários. Desta forma, a Fase 1 deste trabalho foi dedicada exclusivamente à análise desta característica, de modo que, a execução desta etapa complementar auxiliou na maturidade da pesquisa, com o intuito de extrair os melhores resultados nos passos seguintes. A característica de usuários hiperespecialistas foi escolhida devido ao fato de já ter sido reportado pela literatura. Deste modo, era sabido que existia a possibilidade de encontrarmos usuários com essa característica. Além disso, esse é um importante perfil de usuário, pois desenvolvedores com esse perfil beneficiam o crescimento do *crowdsourcing*, pois tornam real a possibilidade de as empresas contarem com um *pool* global de especialistas a um valor reduzido, já que nem sempre é possível encontrar profissionais com habilidades específicas na região, ou o alto custo pode inviabilizar a contratação dos mesmos. Para complementar este trabalho, a fase da análise do perfil de contribuição foi proposta para explorar outras formas de classificar os perfis de contribuição. As duas fases utilizam os dados coletados na seção 3.3, porém, para cada uma delas, definiu-se escopos e formas diferentes para o uso dos dados. Desse modo, os detalhes sobre a coleta e armazenamento dos dados estão descritos a seguir, bem como a definição em alto nível de cada fase. Os detalhes de cada fase estão descritos nos capítulos 4 e 5.

Figura 3.1: Representação geral do método



### 3.1 Hiperespecialização

A Fase 1 (Hiperespecialização), descrita em detalhes no Capítulo 4, tem o intuito de aprofundar os estudos relacionados a um perfil de contribuição específico e previamente determinado: os hiperespecialistas (Malone et al., 2011). Essa fase tem por objetivo responder à questão **QP1**, que visa descobrir mais informações sobre os usuários classificados como hiperespecialistas. A fase é composta por duas etapas distintas que se complementam na busca por investigar essa característica formadora de perfil de contribuição.

### 3.2 Perfil de contribuição

A Fase 2 (Perfil de contribuição) não foca em um determinado perfil, mas consiste em identificar e catalogar diferentes perfis de contribuição dos usuários da plataforma

Topcoder. Essa fase tem por objetivo responder à questão de pesquisa **QP2**. Os detalhes sobre essa fase estão disponíveis na Seção 5.

### 3.3 Coleta dos dados

A coleta dos dados consistiu-se de buscar informações relacionadas ao perfil de contribuição e ações dos participantes da plataforma Topcoder. Primeiramente, foi necessário identificar os dados que estariam disponíveis para coleta. Para isso observou-se que seria possível coletar as informações necessárias via APIs da própria plataforma. Desta maneira, foi selecionado um conjunto de APIs que retornariam parte das informações disponíveis sobre as ações realizadas pelos usuários, bem como os dados dos desafios que eles participaram. É importante ressaltar que os desafios na plataforma são distribuídos em três categorias distintas, sendo elas Desenvolvimento, *Design* e Ciência de Dados. A categoria Ciência de Dados possui desafios de dois subtipos: SRMs (*Single Round Matches*) e *marathon*.

As *Uniform Resource Locators* (URLs) utilizadas para a coleta dos dados são exemplificadas abaixo onde a palavra “[member]” representa um nome de usuário válido:

- [api.topcoder.com/v2/users/\[member\]/](https://api.topcoder.com/v2/users/[member]/): busca informações pessoais como país de origem, a data que o usuário se cadastrou, as conquistas atingidas e se o usuário é copiloto;
- [api.topcoder.com/v3/members/\[member\]/](https://api.topcoder.com/v3/members/[member]/): completa a busca por informações pessoais como, por exemplo, identificador do usuário e os tipos de desafios de que o usuário diz participar;
- [api.topcoder.com/v3/members/\[member\]/skills/](https://api.topcoder.com/v3/members/[member]/skills/): pesquisa as habilidades que o usuário diz possuir;
- [api.topcoder.com/v3/members/\[member\]/challenges/?filter=track%3dDEVELOP](https://api.topcoder.com/v3/members/[member]/challenges/?filter=track%3dDEVELOP): pesquisa os desafios do tipo desenvolvimento de que os usuários participaram;
- [api.topcoder.com/v3/members/\[member\]/challenges/?filter=track%3dDESIGN](https://api.topcoder.com/v3/members/[member]/challenges/?filter=track%3dDESIGN): pesquisa os desafios do tipo *design* de que os usuários participaram;
- [api.topcoder.com/v3/members/\[member\]/srms/](https://api.topcoder.com/v3/members/[member]/srms/): pesquisa os desafios do tipo SRM de que os usuários participaram;

- [api.topcoder.com/v3/members/\[member\]/mms/?filter=track%3dDEVELOP](https://api.topcoder.com/v3/members/[member]/mms/?filter=track%3dDEVELOP): busca os desafios do tipo *marathon* de que os usuários participaram.

As APIs citadas anteriormente foram utilizadas para a coleta das informações, contudo, um ponto a ser levado em consideração é a necessidade do nome do usuário para que elas possam retornar as informações relacionadas ao membro. Dessa maneira, foram propostas três abordagens para que nomes de usuários da plataforma pudessem ser coletados:

1. coleta, *scrapping* na página de busca por país da aplicação<sup>1</sup>. Esta abordagem não foi completamente efetiva devido ao fato de, para alguns países, algumas páginas não carregarem corretamente, de modo que o conteúdo delas não foi exibido.
2. coleta a partir de uma API<sup>2</sup> que contém os membros que já participaram de SRMs. Esta abordagem não pode ser muito explorada devido ao fato de que algumas páginas subsequentes não são carregadas.
3. com parte dos dados dos desafios coletados, foi possível utilizar outra API<sup>3</sup> que retorna, entre outros dados, o nome de usuário de todos os participantes registrados em um desafio. Como já haviam sido coletados alguns desafios, foi possível usar esses identificadores para buscas novos usuários.

O processo de busca de nomes de usuários foi repetido por várias vezes, e com todas as abordagens anteriormente citadas, para que fosse atingido o número máximo de participantes possível. Enquanto o processo de coleta de nomes estava ativo, foi iniciada de maneira paralela a busca pelas informações dos participantes cujos nomes já haviam sido coletados. Este processo se estendeu por vários meses para que fosse possível atingir o maior número possível de participantes cadastrados na plataforma.

A coleta de informações teve início no primeiro dia do mês de agosto de 2016, sendo finalizada no dia 1 de dezembro de 2017. Atualmente o banco de dados tem armazenado todas as participações nos diversos tipos de desafios de cerca de 350.000 usuários da plataforma. Alguns dados adicionais foram coletados, como a nacionalidade, data do cadastro, habilidade que os usuários julgavam possuir e as conquistas que os membros obtiveram durante sua jornada.

É importante notar que, ao longo da coleta, foram identificadas algumas inconsistências nos dados dos usuários. Basicamente foram identificados dois tipos de problemas: (i) para alguns usuários não foi possível iterar através de todas as páginas de

<sup>1</sup><https://www.topcoder.com/tc?module=ViewAdvanced>

<sup>2</sup><http://api.topcoder.com/v2/users/tops/data?pageIndex=1&pageSize=40>

<sup>3</sup>[api.topcoder.com/v2/challenges/registrants/\[identificador\]](https://api.topcoder.com/v2/challenges/registrants/[identificador])

desafios; e (ii) alguns usuários não tiveram dados disponíveis para coleta, porque eles encerraram suas contas. Um total de 39 usuários com esses problemas foram descartados das análises futuras.

### 3.4 Modelo dos dados

Os dados foram armazenados em um banco de dados relacional cujo sistema gerenciador é o MySQL. O MySQL foi utilizado por ser um banco de dados de código aberto e amplamente difundido em todo o mundo. Atualmente o tamanho da base é de 535 megabytes e seu diagrama de entidade e relacionamento (DER) está disponível no Apêndice B.

A principal tabela do modelo é a tabela “Membro”. Ela representa cada um dos 350 mil usuários coletados. Esta tabela possui os seguintes atributos:

- Identificador local do usuário (chave primária);
- Identificador do usuário na plataforma;
- Apelido do usuário (nome utilizado na plataforma);
- Data do cadastro;
- País de origem;
- Copiloto (sim ou não);
- Linguagem de programação para os desafios de SRM (quando informado);
- Data da última alteração na conta;
- Contadores para cada um dos tipos de desafios indicando a quantidade de desafios que o usuário participou.

O atributo copiloto indica se o usuário está apto para mediar, junto à plataforma, as ações das empresas clientes, ou seja, este usuário é um membro experiente da comunidade que auxilia as empresas durante o ciclo da tarefa. Além desses atributos, a classe “Membro” possui relações com algumas outras tabelas. Esses relacionamentos podem ser de dois tipos: “um para muitos” (*one to many*), ou “muitos para muitos” (*many to many*). A relação “um para muitos” indica que o usuário pode ter um conjunto de registros, sendo que esse registro é somente do usuário. Neste caso, o registro armazena a



chave primária do membro como chave estrangeira em sua tabela. A relação “muitos para muitos” indica que os usuários podem ter uma coleção de registros de uma tabela e esses registros podem estar associado a vários usuários. Nesse caso, uma tabela associativa com as chaves dos dois registros foi criada. A Tabela - 3.1 mostra todos os relacionamentos da tabela “Membro”. Cada item citado na Tabela - 3.1 será devidamente explicado posteriormente.

**Tabela 3.1: Relacionamentos da tabela “Membro”**

Nome da tabela	Tipo de Relacionamento
<i>Tracks</i>	<i>Many to many</i>
Habilidades	<i>Many to many</i>
Conquista	<i>One to many</i>
MembroDesafioDesenvolvimento	<i>One to many</i>
MembroDesafioDesign	<i>One to many</i>
MembroSRM	<i>One to many</i>
MembroMaratona	<i>One to many</i>
Fórum	<i>One to many</i>
Postagem	<i>One to many</i>

A próxima tabela a ser mencionada é a “Track”. Esta é simples, pois ela somente possui três registros, sendo eles: *Develop*, *Design* e *Data Science*. O usuário pode escolher entre esses três registros para indicar que o mesmo é desenvolvedor atuante em desafios de desenvolvimento (*Develop*), *design* e/ou Ciência de Dados (*Data Science*). Desse modo, existe uma tabela associativa que armazena o código do membro e o código dos seguimentos escolhidos pelo usuário.

A tabela “Habilidade” é semelhantes à “Track”, pois é opção do usuário informar todas as habilidades que o mesmo julga possuir. E assim como na tabela anterior, existe outra tabela que armazena os identificadores dos membros e os códigos das habilidades. Atualmente esta tabela conta com 316 habilidades distintas.

A tabela “Conquista” indica as marcas atingidas pelas usuários. Por exemplo, quando um usuário vence seu primeiro desafio, ou quando ele posta pela primeira vez no fórum de discussão, o usuário atinge uma marca e recebe um distintivo. A Figura - 3.2 representa graficamente as conquistas que um usuário atingiu.

As tabelas “Plataforma” e “Tecnologia” são semelhantes e auxiliam na descrição dos desafios. Ambas indicam tecnologias que devem ser utilizadas no desenvolvimento dos artefatos, porém a diferença entre elas é que a tabela “Tecnologia” tende a fornecer opções como linguagens de programação, *frameworks* e sistemas gerenciadores de banco de dados. Enquanto a tabela “Plataforma” tende a fornecer opções como sistemas operacionais,

**Figura 3.2:** Exemplo das conquistas de um usuário extraído da plataforma Topcoder



plataformas de desenvolvimento de software ou plataformas de hospedagem de sistemas. Contudo existem informações que acabam se cruzando e ficam nas duas tabelas, como por exemplo, Node.JS, HTML e Android. Essas tabelas são utilizadas exclusivamente pelos desafios do tipo desenvolvimento e *design*, mas não existe a obrigatoriedade dos desafios informarem esses dois atributos, visto que existem desafios que não possuem essas informações. No início da coleta, as informações das duas tabelas eram mencionadas nas páginas dos desafios, contudo, atualmente a tabela “Plataforma” não é utilizada na página do desafio.

As tabelas “DesafioDesenvolvimento”, “DesafioDesign”, “Srm” e “Maratona” são utilizadas para armazenar as definições dos desafios correspondentes. As tabelas “DesafioDesenvolvimento” e “DesafioDesign” possuem relacionamento de “muitos para muitos” com as tabelas “Plataforma” e “Tecnologia”. A outra relação é com a tabela “membro”, porém essa relação não é feita diretamente e sim por intermédio das tabelas “MembroDesafioDesenvolvimento” e “MembroDesafioDesign”, que serão abordadas futuramente. As tabelas “Srm” e “Maratona” possuem a relação com a tabela “Rounds”, indicando que cada desafio pode ter várias etapas. Assim como nos desafios de desenvolvimento e *design*, essas tabelas não têm relação direta com o membro. Desse modo, elas possuem relação com “MembroSrm” e “MembroMaratona”.

A tabela “MembroDesafioDesenvolvimento” é responsável por armazenar a relação entre desafios do tipo desenvolvimento com os usuários que participaram deles. Caso o usuário tenha submetido no desafio, um registro é adicionado na tabela “SubmissaoDesenvolvimento”. É possível que o usuário tenha várias submissões no mesmo desafio,

deste modo cada submissão é arquivada, contudo, somente a última submissão é válida, sendo a única a ser analisada pela plataforma. Caso o usuário possua uma submissão vitoriosa, um registro foi inserido na tabela “SubmissaoDesenvolvimentoVencedora”. Esse registro indica qual das submissões do usuário é a escolhida pelo responsável da tarefa. A última relação da tabela “MembroDesafioDesenvolvimento” é com a tabela “Regra”. Esta indica quais os tipos de papéis que os usuários podem desempenhar nos desafios. Na maioria dos casos, a regra definida é a de competidor (*submitter*), essa regra indica que o usuário participa dos desafios para poder concorrer à recompensa. Contudo existem outras regras como revisores, copilotos e observadores. Desse modo existe uma tabela associativa que armazena os identificadores de “MembroDesafioDesenvolvimento” e “Regra” para representar qual era o papel do usuário no desafio de que ele participou.

A tabela “MembroDesafioDesign” é similar à tabela “MembroDesafioDesenvolvimento”. Ela armazena as participações do usuário nos desafios de design. Ela também possui a relação entre as submissões e as submissões vitoriosas que cada usuário fez nos desafios de que ele participou. Assim como nos desafios do tipo desenvolvimento, cada usuário participa dos desafios de *design* com uma regra. Contudo essa é uma informação que não foi coletada até o momento.

A tabela “MembroSrm” associa as participações dos usuários nos desafios classificados como SRM. Além disso, essa tabela possui um relacionamento para armazenar as variações de pontuação ao longo das rodadas em cada participação dos usuários nos desafios. Essa relação é feita por meio da tabela “DetalheMembroSrm”, que armazena a antiga e a nova pontuação, a colocação no round, o total de pontos e a colocação na divisão. Essa é a informação mais próxima às submissões feitas nos desafios de desenvolvimento e *design* que é disponibilizada pelas APIs.

A tabela “MembroMaratona” funciona de modo similar à tabela “MembroSrm”, a diferença é que ela armazena dados das maratonas participadas pelos usuários. Assim como a tabela “MembroSrm”, esta tabela também possui as informações relacionadas à participação dos usuários em cada rodada das maratonas participadas.

Por fim, a tabela “Membro” possui relacionamento com as tabelas “Fórum” e “Postagem”. A tabela “Fórum” indica um tópico do fórum. A tabela “Postagem” representa as respostas que o tópico obteve. As duas tabelas possuem relação com o membro devido ao fato de o mesmo realizar as ações de criar um novo tópico ou responder a um tópico. A tabela “Postagem” tem a relação com “Fórum” porque toda a postagem deve ser de um tópico. Esses dados não foram coletados ainda, pois eles não estão disponíveis em nenhuma API pública, sendo necessário usar uma estratégia de *scrapping* nas páginas dos

fórums. O algoritmo foi desenvolvido e está apto para fazer essa coleta, contudo, ao longo deste trabalho esses atributos não foram requeridos. Desse modo à coleta não foi iniciada.

---

## Fase da Hiperespecialização

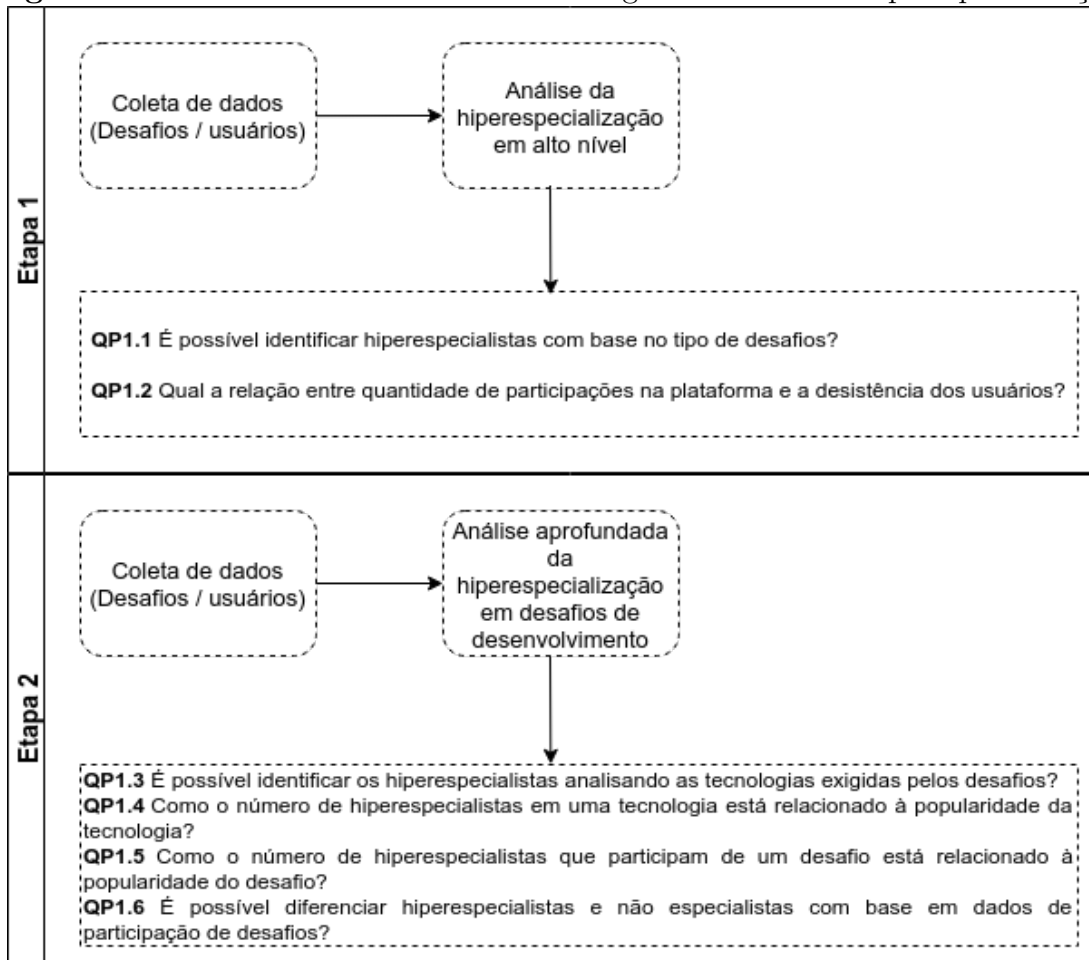
---

A fase que busca estudar o fenômeno da Hiperespecialização foi dividida em duas etapas. A Figura - 4.1 apresenta uma visão de alto nível do método seguido. Na Etapa 1 foi realizado um estudo exploratório para analisar o fenômeno dos hiperespecialistas em alto nível, verificando o perfil de contribuição dos participantes em termos de tipos de desafios escolhidos. Os resultados da Etapa 1 indicam que foi possível identificar a manifestação da hiperespecialização. Na Etapa 2 foi realizada uma análise aprofundada dos desafios de desenvolvimento – focando em tarefas de competição com recompensas financeiras. Cada etapa envolveu sua própria filtragem e análise de dados. Desse modo, a Seção 4.1 apresenta todos os detalhes da primeira etapa desta fase. Na Seção 4.2, é descrita a etapa aprofundada da fase da hiperespecialização.

### 4.1 Etapa 1: Análise preliminar do perfil de hiperespecialista

O objetivo desta etapa foi explorar preliminarmente o fenômeno da hiperespecialização em *crowdsourcing* para desenvolvimento de software. Nesta etapa, investigou-se a possível manifestação da hiperespecialização em um amplo contexto. Para isso, analisou-se a participação dos usuários da plataforma Topcoder de acordo com o tipo de desafio desenvolvido ao longo do tempo. Os detalhes sobre o método são apresentados a seguir.

**Figura 4.1:** Visão de alto nível do método seguido na fase da Hiperespecialização



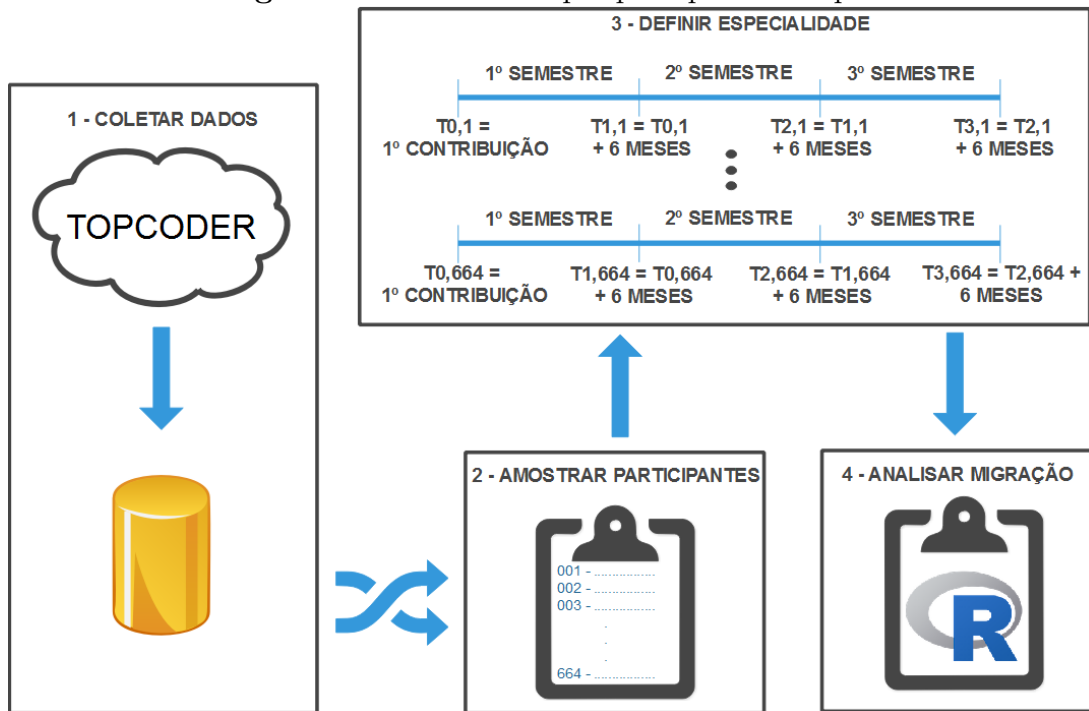
#### 4.1.1 Método de pesquisa

A questão de pesquisa central analisada nesta seção é a **QP1** *Como o perfil de contribuição conhecido como “hiperespecialista” (Malone et al., 2011) pode ser observado na plataforma Topcoder?*. E para auxiliar nesta investigação, duas questões de pesquisa específicas foram abstraídas da questão da **QP1**, sendo elas:

- **QP1.1** É possível identificar hiperespecialistas com base no tipo de desafios?
- **QP1.2** Qual a relação entre quantidade de participações na plataforma e a desistência dos usuários?

O método seguido nesta etapa para responder às questões de pesquisa é apresentado na Figura - 4.2. A coleta (passo 1) foi descrita na Seção 3.3, enquanto a filtragem de dados, a classificação dos hiperespecialistas e a análise dos resultados (passos 2-4) são especificadas nas subseções abaixo.

Figura 4.2: Método de pesquisa para a Etapa 1



#### 4.1.2 Filtragem e análise dos dados

No passo 2 (Figura - 4.2), a população amostral foi definida seguindo estes critérios: (i) os usuários deveriam ter participado de pelo menos um desafio (os usuários deveriam ter, ao menos, se inscrito para participar do desafio, não, necessariamente ter enviado contribuições); e (ii) a data de abertura do desafio deveria ser pelo menos 18 meses antes do início da coleta de dados, já que esse foi o período de tempo que os usuários foram analisados. Entre os 92.263 usuários que atendiam aos critérios, 664 usuários foram amostrados aleatoriamente. O tamanho da população de amostragem foi definido com um nível de confiança de 99% e uma margem de erro de 5%.

No passo 3, foi contabilizado o número de desafios de que cada usuário participou. Depois disso, foi analisado se a hiperespecialização poderia ser observada considerando os tipos de tarefas escolhidas pelos usuários. O objetivo desse passo foi realizar uma análise temporal para cada usuário, verificando o número de desafios de que cada usuário participou, de acordo com o tipo da tarefa. Essa classificação serviu de base para a análise da existência de hiperespecialistas ao longo do tempo. Para isso, foi definida a linha de tempo de cada participante, considerando a data em que os usuários participaram em seu primeiro desafio. A linha do tempo foi definida para 18 meses, calculados a partir da data

do primeiro desafio. Para analisar a hiperespecialização, esse período foi dividido em três períodos de 6 meses, conforme descrito na Figura - 4.2.

Usando a linha de tempo dos usuários, cada um foi classificado como hiperespecialista ou não especialista para cada período analisado. Foi definido que um participante seria considerado como hiperespecialista se pelo menos 75% dos desafios em que o desenvolvedor participasse fossem do mesmo tipo. Se o limite de 75% não foi alcançado, o participante foi classificado como “não especialista”. Além disso, os usuários que não participaram de qualquer desafio em um determinado período de 6 meses foram classificados como “nenhuma contribuição”. Esse fato ocorreu apenas no segundo ou terceiro período de análise, já que foram amostrados apenas desenvolvedores com pelo menos um desafio. O limite de 75% foi definido pelos autores, uma vez que não foi possível encontrar valores da literatura que poderiam ser utilizados para esse fim. Como esta era uma etapa preliminar, assume-se que esse valor é justo para estudar o fenômeno, pois esse é baseado na distribuição dos desenvolvedores na plataforma. Além disso, entendemos que isso pode representar uma ameaça à validade.

No passo 4, a classificação do usuário de acordo com a hiperespecialização foi analisada nos três períodos de tempo. Foi verificado se houve uma “mudança” ou a “manutenção” da especialidade dos participantes comparando o período inicial de 6 meses com os seguintes períodos. Por meio desta comparação, foi realizada uma análise descritiva dos dados para verificar se o fenômeno de hiperespecialização foi observado no contexto deste trabalho.

Também foi verificado se existe variação da quantidade de participações dos usuários que competiram ao longo dos três períodos. Para essa análise, foi utilizado o teste estatístico ANOVA *One-Way Repeated Measure*, a fim de comparar três ou mais resultados oriundos da observação do mesmo grupo de amostras. No contexto deste teste, foram selecionados todos os usuários que participaram nos três períodos (incluindo os não especialistas). Testou-se a seguinte hipótese nula ( $H_0$ ): *a participação de desafios nos três períodos é igual em relação ao número de desafios participados.*

O Teste Chi-Square foi utilizado para avaliar a existência de relação entre o baixo número de desafios com o abandono ou a permanência na plataforma. Para esse teste foram criados dois grupos. O primeiro composto da quantidade de desafios do primeiro semestre dos *participantes que continuaram* na plataforma no segundo semestre; o segundo é composto pelas participações do primeiro período dos *usuários que abandonaram* a plataforma (não participaram de nenhum desafio) no segundo semestre. A hipótese nula ( $H_0$ ) é: *a quantidade de participação nos desafios não está associada à permanência desistência dos usuários na plataforma.*



### 4.1.3 Resultados

Esta seção apresenta os resultados da Etapa 1, organizados de acordo com as questões de pesquisa.

#### QP1.1 É possível identificar os hiperespecialistas com base no tipo de desafios?

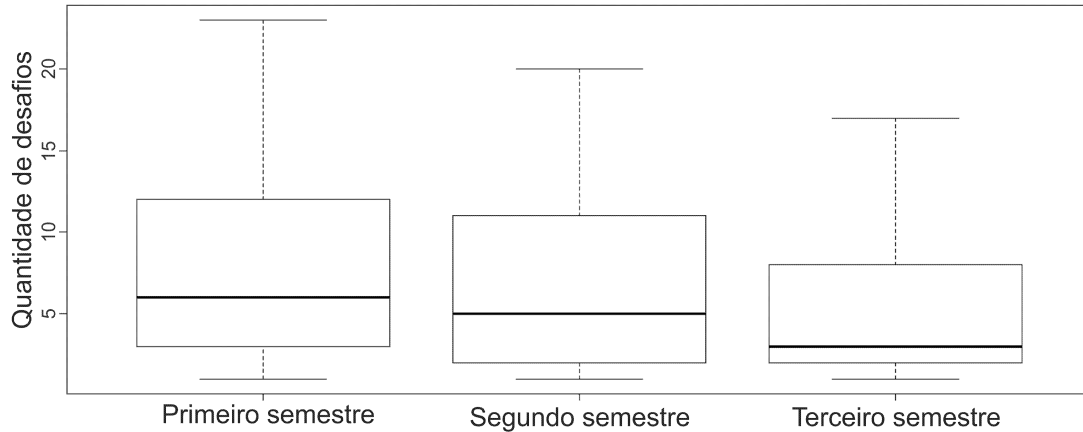
Dentre os 664 participantes analisados, apenas 98 (14% da amostra) continuaram contribuindo ao longo dos três períodos (18 meses) do estudo. Destes 98 participantes, 92 (93,8% dos participantes que permaneceram) continuaram contribuindo no mesmo tipo de desafio ao longo do período de 18 meses. Esse resultado indica que a hiperespecialização pode se manifestar em relação ao tipo de desafio de que o usuário participa. O número de hiperespecialistas nos três períodos, de acordo com o tipo de desafio, foi: 8 usuários hiperespecialistas em desenvolvimento, 4 em design e 80 em Ciência de Dados.

Observou-se ainda que 35 participantes estavam ausentes no segundo período de análise, mas retornaram no terceiro. Entre esses participantes, apenas 3 (8,6%) mudaram sua especialidade do primeiro para o terceiro período - todos foram classificados como especialistas em Ciência de Dados no primeiro semestre e, no terceiro, foram classificados como especialistas em desenvolvimento (2) e *design* (1). Todos os outros 32 (91,4%) foram classificados com a mesma especialidade em ambos os períodos.

Uma característica em relação aos usuários que se mantiveram contribuindo ao longo dos três períodos é a diferença na quantidade de contribuições realizadas por cada usuário. Para analisar essa característica, utilizou-se o teste ANOVA, comparando a quantidade de desafios disputados por cada usuário nos três períodos. O resultado mostra que há uma diferença no número de desafios ( $F=6,07$ ;  $p\text{-value}=0,003$ ), rejeitando  $H_0$ . Os resultados das comparações múltiplas –  $p\text{-values}$  foram ajustados usando o *Tukey method* – mostraram que o número de desafios disputados no primeiro período é diferente do segundo período ( $t\text{-ratio}=2,48$ ;  $p\text{-value}=0,04$ ). Houve também uma diferença no número de desafios entre o primeiro e terceiro períodos ( $t\text{-ratio}=3,36$ ;  $p\text{-value}=0,002$ ). No entanto, não há evidências de que os valores para o segundo e terceiro períodos sejam diferentes ( $t\text{-ratio}=0,881$ ;  $p\text{-value}=0,653$ ). Por meio dessa análise, foi possível verificar que os usuários que contribuem em todos os períodos têm o pico de contribuição no primeiro semestre, reduzindo o número de contribuições no segundo período, que permanecem constantes no terceiro período (Figura - 4.3).

#### QP1.2 Qual a relação entre quantidade de participações na plataforma e a desistência dos usuários?

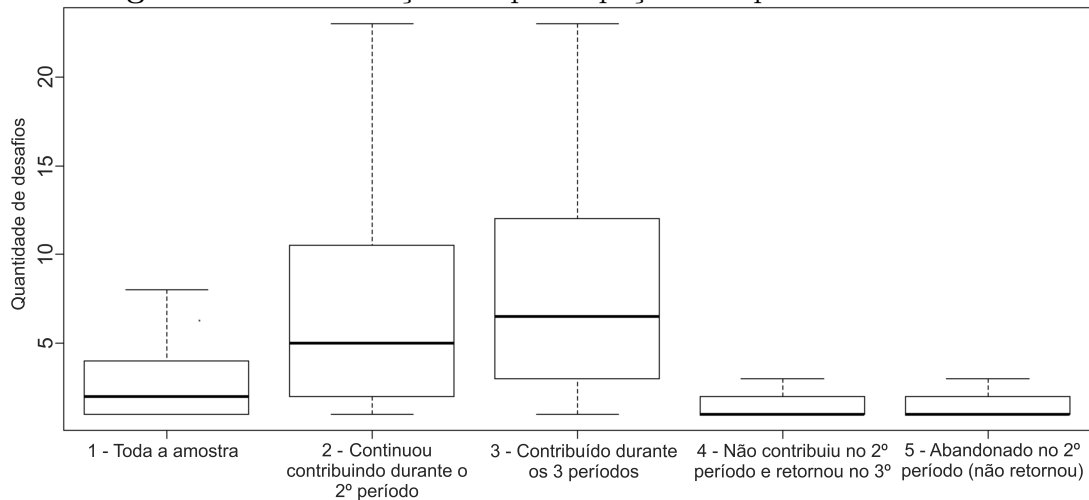
**Figura 4.3:** Distribuição do número de desafios participados pelos usuários nos períodos analisados



O fato de cerca de 66% dos desenvolvedores da amostra possuírem contribuições apenas no primeiro período analisado (seis primeiros meses), indica o possível abandono da plataforma. Esse fato corrobora os resultados apresentados por (Zanatta et al., 2017), em que os autores apresentam as barreiras que os recém-chegados enfrentam ao tentar participar de desafios na plataforma. Nesta etapa, observou-se que, dos 441 usuários que pararam de contribuir, 272 participaram de apenas um desafio. Para os outros 169 usuários que desistiram, houve uma variação entre duas e 28 participações (mediana = 3, desvio padrão = 3,44). Essa característica pode indicar pelo menos cinco possíveis situações: (i) os usuários não se adaptaram aos padrões da plataforma (dificuldade de encontrar tarefas adequadas, problemas de interação com a plataforma e com outros usuários, entre outros); (ii) falta de conhecimento para completar tarefas; (iii) ineficiência dos métodos de treinamento proposto pela plataforma; (iv) os usuários já atingiram seu objetivo – participar/treinar em alguma tecnologia específica ou ganhar uma determinada quantia de dinheiro; ou (v) os usuários estavam procurando ganhar dinheiro rapidamente e não venceram os primeiros desafios - investindo seu tempo sem o esperado “retorno do investimento”.

Na Figura - 4.4 é possível observar a distribuição da quantidade de participação da amostra no primeiro período (os *outliers* não são apresentados para uma melhor visualização). Observando os *boxplots*, é possível notar que, em geral, os usuários que param de contribuir (abandonam) participam de menos desafios do que os usuários que permanecem ativos na plataforma nos períodos seguintes.

Para validar a análise dos *boxplots*, e verificar se existe uma relação entre a quantidade de desafios participados e o abandono/permanência na plataforma, foi realizado o teste Chi-Square. O resultado do teste mostra que o número de desafios de que o

**Figura 4.4:** Distribuição das participações nos períodos verificados

usuário participou é um indicativo de abandono/permanência, rejeitando  $H_0$  ( $X^2=197,18$ ,  $p\text{-value}=0,001$ ). No entanto, esta é apenas uma análise preliminar. Outros tipos de estudos podem ser realizados para entender melhor esse fenômeno.

## 4.2 Etapa 2: Hiperespecialização em desafios de desenvolvimento

Dado os promissores resultados da Etapa 1, em que foi possível evidenciar o fenômeno da hiperespecialização em termos do tipo de desafio, na Etapa 2, decidiu-se explorar o fenômeno da hiperespecialização em um contexto mais específico. Foi definido que os esforços seriam concentrados nos desafios classificados como “desenvolvimento”. Esse tipo específico foi escolhido, uma vez que esses desafios são propostos pelas empresas (em oposição aos desafios do tipo Ciência de Dados, que são principalmente relacionados a desafios parecidos com as maratonas) e oferecem recompensas financeiras. Assim, o objetivo nesta etapa foi caracterizar os hiperespecialistas no contexto dos desafios de desenvolvimento com recompensa financeira na plataforma Topcoder.

### 4.2.1 Método de pesquisa

Seguiu-se analisando a questão de pesquisa central desta fase (**QP1**. *Como o perfil de contribuição conhecido como “hiperespecialista” (Malone et al., 2011) pode ser observado na plataforma Topcoder?*). Nessa etapa foram definidas outras quatro questões de pesquisa específicas:

- **QP1.3** É possível identificar os hiperespecialistas analisando as tecnologias exigidas pelos desafios?
- **QP1.4** Como o número de hiperespecialistas em uma tecnologia está relacionado à popularidade da tecnologia?
- **QP1.5** Como o número de hiperespecialistas que participam de um desafio está relacionado à popularidade do desafio?
- **QP1.6** É possível diferenciar hiperespecialistas e não especialistas com base em dados de participação de desafios?

A QP1.4 busca analisar a correlação entre a quantidade de desafios que requerem as tecnologias e a quantidade de hiperespecialistas que submeteram nestes desafios, ou seja, analisar a correlação entre popularidade da tecnologia e quantidade de hiperespecialistas. A QP1.5 busca verificar se existe correlação entre a quantidade de usuários que os desafios atraem comparado com a quantidade de hiperespecialistas em tecnologias que o desafio requer, ou seja, analisar a correlação entre a popularidade do desafio com a quantidade de hiperespecialistas.

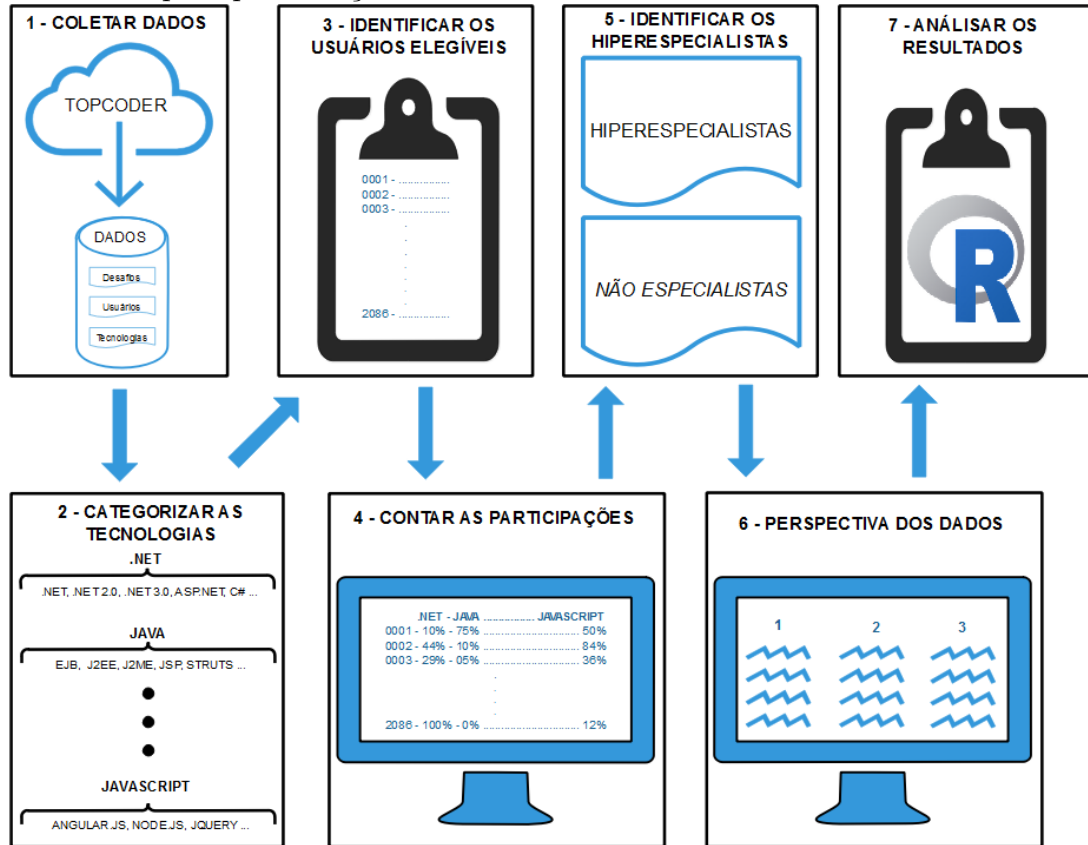
Para responder a essas perguntas, seguiu-se o método apresentado pela Figura - 4.5. Coleta e seleção de dados (passos 1-4) e análise (passos 5-7) são especificadas nas subseções abaixo.

#### **4.2.2 Coleta e seleção de dados**

Inicialmente, a base contava com dados de 29.276 usuários que participaram de algum desafio do tipo desenvolvimento e contava com 15.351 desafios distintos. Para cada usuário foram utilizados dados relacionados à sua participação em desafios, incluindo o papel desempenhado (concorrente, revisor, copiloto), colocação em cada desafio e informações sobre o número de submissões. Quando o usuário possui alguma submissão que resultou em uma recompensa financeira, chamada de submissão vencedora, essa informação também foi armazenada. Além disso, dados sobre desafios como o total de recompensas, o início do registro e as tecnologias necessárias também foram coletados e armazenados.

Foi possível identificar a existência de 200 tecnologias diferentes associadas aos desafios coletados. No entanto, examinando essas tecnologias, ficou evidenciada a existência de tecnologias semelhantes ou idênticas representadas por nomenclaturas diferentes, bem como versões específicas de uma tecnologia e estruturas específicas relacionadas a

**Figura 4.5:** Representação gráfica do método utilizado na Etapa 2 da análise da hiperespecialização



tecnologias específicas. Assim, no passo 2, as tecnologias coletadas foram classificadas em categorias mais genéricas para reduzir o conjunto inicial a ser analisado. Essa classificação foi realizada em reuniões dedicadas a esse fim. Usou-se uma abordagem baseada em *card sorting*, na qual discutiu-se cada tecnologia e como classificá-la. O processo foi iterativo até que houvesse um consenso sobre a classificação. Exemplos de categorização incluem: C#, .NET, .NET 2.0 e .NET 3.0 foram classificados como .NET; J2EE, J2SE, Java foram classificados como Java. Ficou claro que essa ação resultaria em uma perda em termos de granularidade. No entanto, trabalhar com tecnologias específicas dificultaria a análise do fenômeno. A lista completa de categorias e tecnologias e suas classificações está disponível neste endereço eletrônico<sup>1</sup>.

No passo 3 (Figura - 4.5), foram selecionados os participantes com base em alguns critérios. Os critérios usados na Etapa 2 são mais restritivos que na Etapa 1, uma vez que o objetivo aqui é analisar toda a vida útil dos usuários na plataforma. Assim, foram filtrados os usuários que registraram e enviaram soluções para, pelo menos, três desafios diferentes.

<sup>1</sup><https://zenodo.org/record/1169411>

O limiar de 3 desafios foi definido após analisar a distribuição do número de desafios em que os usuários coletados participaram, e 3 foi o delimitador do terceiro quartil. Essa ação foi tomada para escolher apenas os usuários que realmente interagiram com a plataforma. É importante deixar explícito que foram considerados apenas os usuários que participaram como “*submitters*” em três desafios, isto é, os usuários que enviaram uma solução proposta (*artefato*) para o desafio. Assim, foram descartadas as pessoas da equipe do Topcoder (por exemplo, revisores e copilotos) e usuários que se registraram para o desafio, mas não enviaram uma solução. Em relação aos desafios, foram considerados apenas os desafios de desenvolvimento que ofereceram uma recompensa financeira; iniciaram o registro antes de primeiro de setembro de 2016; tinham pelo menos uma tecnologia necessária listada e foram completados, ou seja, qualquer desafio que por ventura tenha sido cancelado não foi considerado válido. Por fim, os subtipos de desafios de desenvolvimento não foram considerados nesta fase do estudo, ou seja, o limiar de 3 desafios é baseado apenas na classificação geral do desafio que é desenvolvimento, sem levar em consideração qualquer subclassificação.

Depois de aplicar os critérios para selecionar usuários e desafios, uma população de 2.086 usuários foi considerada na análise. Esses usuários foram considerados no passo 4, no qual foi definida a taxa de participação de cada usuário por tecnologia. Essa taxa foi calculada contando o número de desafios que exigiam uma determinada tecnologia na qual o usuário havia enviado uma solução, dividido pelo número total de desafios em que o usuário enviava soluções independentemente da tecnologia. Por exemplo, se JohnDoe participou de 10 desafios, em 8 deles Java era necessário e, em quatro deles, HTML era necessário, a “taxa de HTML” para JohnDoe é de 40% e a “taxa de Java” é 80%.

### 4.2.3 Análise de dados

Para classificar os desenvolvedores em hiperespecialistas e não especialistas (passo 5), também foi utilizado o limite de 75% usado na Etapa 1. Especialistas foram definidos como aqueles usuários que tiveram pelo menos 75% de seus envios em desafios que exigiram, no mínimo, uma tecnologia em comum (taxa de tecnologia). Usando o exemplo de JohnDoe mais uma vez, desde que sua taxa em Java é de 80%, o JohnDoe é classificado como um especialista em Java. Como mencionado anteriormente, consideramos 75% um limiar justo para analisar este fenômeno.

No passo 6 (Figura - 4.5), os dados foram analisados a partir de três perspectivas (i) tecnologias; (ii) desafios; e (iii) participantes. Do ponto de vista das tecnologias (i), foi analisado: o número de desafios em que cada tecnologia era necessária; o número total de

participantes submetidos aos desafios em que foi exigido; e o número total de especialistas nessa tecnologia. Do ponto de vista dos desafios (ii), foi considerado: o número total de participantes; e o número de especialistas que se submeteram a cada desafio. Finalmente, na perspectiva do participante (iii) foi analisado, para cada participante: o número total de desafios; o número de tecnologias com as quais eles tiveram contato; e o número total de vitórias (número de desafios que o usuário estava em uma veiculação que garantia recompensa financeira). No caso dos hiperespecialistas, foi analisado também o número de desafios com submissões e o número de vitórias nos desafios que continham alguma tecnologia na qual o usuário era um especialista.

O conjunto de dados apresentado anteriormente foi utilizado para responder às perguntas de pesquisa anteriormente declaradas. No passo 7 (Figura - 4.5), para responder **QP1.3** e **QP1.6** foi utilizado a estatística descritiva, apoiada por gráficos que apresentam diferentes perspectivas de nossos dados. Em alguns casos, alguns atributos foram usados para comparar hiperespecialistas e não especialistas. Nesses casos, foi utilizado o teste estatístico de Mann-Whitney-Wilcoxon, dado que é um teste não-paramétrico que não requer a suposição de distribuições normais. Também foi feito uso do teste de tamanho de efeito chamado Cliff-delta para comparar os grupos. Para as outras QPs (**QP1.4** e **QP1.5**) foi utilizado o teste de correlação de Spearman, pois, mais uma vez, os dados não seguiram uma distribuição normal. Para **QP1.4**, foi comparado o número de desafios e o número de participantes por tecnologia. Da mesma forma, foi utilizado o teste de correlação na **QP1.5** para comparar o número de participantes e o número de especialistas por desafio.

#### 4.2.4 Resultados

Como na Etapa 1, os resultados são apresentados de acordo com as questões de pesquisa apresentadas anteriormente.

##### **QP1.3 É possível identificar os hiperespecialistas analisando as tecnologias exigidas pelos desafios?**

A resposta é sim. Considerando os 2.086 usuários analisados, 1.256 ( $\approx 60\%$ ) foram classificados como hiperespecialistas em pelo menos uma tecnologia. Entre as 74 categorias de tecnologias analisadas, foram encontrados especialistas em 34. Para as outras 40, nenhum especialista foi encontrado. Na Tabela - 4.1, é possível observar que dos 1.256 usuários catalogados como hiperespecialistas, 1.106 possuem especialização em uma tecnologia, 60 são especialistas em duas tecnologias, 88 em três, e dois usuários foram classificados como especialistas em quatro tecnologias.

**Tabela 4.1:** Distribuição dos hiperespecialistas pelo número de tecnologias nas quais são especializados

# de hiperespecialista	# de tecnologias
1106	1
60	2
88	3
2	4

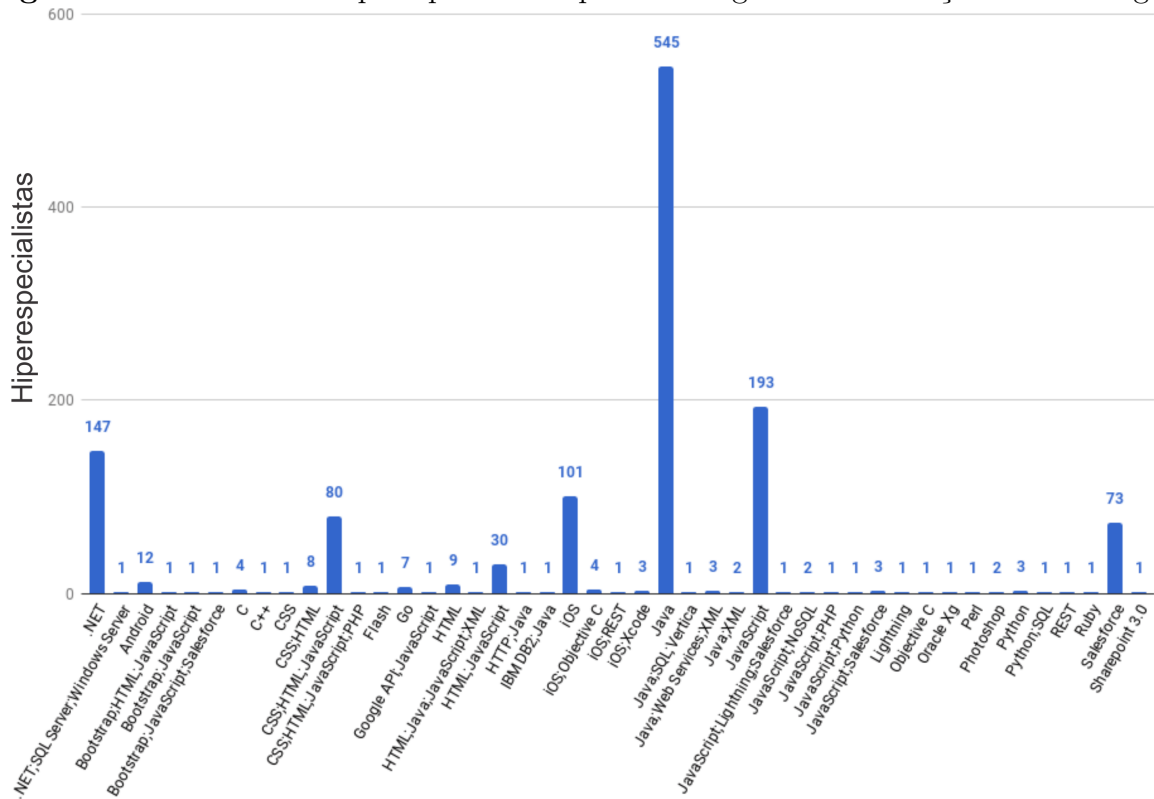
A Tabela - 4.2 mostra as 10 tecnologias com o maior número de hiperespecialistas identificados e o número de desafios que exigiam cada tecnologia. Java, JavaScript e .NET são as três principais em termos do número de hiperespecialistas com 554, 317 e 148, respectivamente. Considerando o número de desafios, é possível observar que as três principais tecnologias são diferentes, já que o HTML apareceu na terceira posição ao invés de .NET. Também é possível perceber que a linguagem Go aparece como necessária em apenas 23 desafios. No entanto, foi possível identificar sete hiperespecialistas nessa tecnologia.

**Tabela 4.2:** Top-10 tecnologias em termos do número de hiperespecialistas disponíveis

Tecnologias	# de hiperespecialistas	# de desafios
Java	554	7097
JavaScript	317	5850
.NET	148	2965
HTML	130	3639
iOS	109	1480
CSS	90	2311
Salesforce	78	783
Android	12	904
Go	7	23
XML	6	717

A Figura - 4.6 traz mais detalhes sobre o número de hiperespecialistas por tecnologia, incluindo a combinação de hiperespecialização observada. Das 45 hiperespecializações apresentadas na figura, 23 apresentam múltiplas tecnologias. Somando o número de hiperespecialistas nessas combinações, foi verificado que 151 (9,1%) usuários apresentam múltiplas especialidades. Entre eles, 89 incluem HTML e CSS (Cascading Style Sheets); sendo: oito especialistas apenas em CSS e HTML, 80 em CSS, HTML e JavaScript, e um em CSS, HTML, JavaScript e PHP. Esses são casos aparentemente extremos de hiperespecialização em que os usuários buscam desafios que exigem um determinado conjunto de tecnologias.



**Figura 4.6:** Número de hiperespecialistas por tecnologia ou combinação de tecnologias

Como mencionado anteriormente, não foi possível identificar hiperespecialistas para 40 tecnologias. Entre eles, é possível verificar tecnologias como Docker, Illustrator, Cobol, Fortran, Ruby on Rails, PostgreSQL e MySQL. Sabe-se que a plataforma Topcoder possui mecanismos para incentivar os usuários a adquirir novos conhecimentos quando há demandas e há pouquíssima mão de obra disponível na plataforma. Esse fato levou a investigar o número de desafios em que essas tecnologias eram requeridas, dados que são apresentados na Tabela - 4.3.

No caso do Docker, Illustrator, Cobol e Fortran, acredita-se que o número de usuários dessas tecnologias pode ser limitado devido à baixa oferta de desafios. No caso dos bancos de dados relacionais MySQL e PostgreSQL, o número de desafios é maior. Analisando os desafios que exigiam essas tecnologias, foi possível identificar que quatro desafios exigiam unicamente o MySQL e um exigia que o PostgreSQL fosse a única tecnologia necessária. Isto pode indicar que os usuários que participaram de desafios que exigiram essas tecnologias, junto com outras, não foram atraídos pelo banco de dados específico, mas por alguma outra tecnologia exigida pelo desafio. Para reforçar essa hipótese, os usuários que participaram dos desafios envolvendo o MySQL e o PostgreSQL foram analisados.

**Tabela 4.3:** Top-10 tecnologias sem hiperespecialista

Tecnologia	# de desafios
MySQL	414
PostgreSQL	279
PhoneGap	127
Flex	77
Commerce Server 2009	72
XSL	54
Ruby on Rails	47
Docker	46
Google App Engine	46
UML	44

Descobriu-se que 56% dos desafios do MySQL e 53% dos desafios do PostgreSQL têm hiperespecialistas em outras tecnologias como PHP, Java e Javascript.

Além disso, investigou-se a existência de hiperespecialistas em bancos de dados relacionais, ao invés de considerar as tecnologias de banco de dados específicas. Para isso, PostgreSQL, MySQL, DB2, SQL Server, Oracle e SQL (Structured Query Language) foram agrupados em uma categoria e considerou-se hiperespecialistas em banco de dados relacional os usuários que envolvessem qualquer uma das tecnologias de BD relacional. Como resultado, foram encontrados sete hiperespecialistas. No entanto, cinco deles foram previamente identificados como hiperespecialistas em tecnologias específicas de banco de dados (banco de dados) (SQL (2), DB2 (1), Oracle (1) e SQL Server (1)). Assim, apenas dois hiperespecialistas foram encontrados agregando as tecnologias de banco de dados. Essa observação reforça a hipótese de que os usuários foram atraídos por outras tecnologias, não por bancos de dados.

#### **QP1.4 O número de hiperespecialistas em uma tecnologia está relacionado à popularidade da tecnologia?**

A resposta curta é sim. Para responder à questão, foi calculada a correlação entre o número de desafios que exigiam uma tecnologia e a porcentagem dos participantes de desafios que envolviam essa tecnologia como hiperespecialistas. A razão (*ratio*) foi calculada para normalizar os dados e analisar a “densidade” dos hiperespecialistas, em vez do número absoluto (conforme apresentado na Tabela - 4.2). O resultado do teste de correlação de Spearman confirmou a observação (0,689). Esse resultado é considerado uma forte correlação positiva (Hinkle et al., 2002), indicando que as tecnologias populares tendem a ter um número maior de hiperespecialistas.

Também foi calculada a correlação usando o número absoluto de hiperespecialistas e o número de desafios que exigem uma tecnologia. O resultado também mostrou

uma correlação muito forte (0,804), mostrando que o número de especialistas está, de fato, correlacionado ao número de desafios disponíveis que exigem a tecnologia que eles dominam.

**QP1.5 O número de hiperespecialistas que participam de um desafio está relacionado à popularidade do desafio?**

A resposta curta é, novamente, sim. Para responder à questão, foi calculada a correlação entre o total de usuários por desafio e o número de hiperespecialistas encontrados no desafio. O resultado encontrado foi 0,456, o que significa correlação positiva moderada, indicando que, quando o número de desenvolvedores aumenta, o número de hiperespecialistas também.

Em um trabalho anterior de Mao et al. (2016), eles mencionam que os membros experientes registram-se cedo nos desafios que dominam para inibir a participação de outros membros. Entretanto, apenas analisando essa correlação, não é possível entender se os resultados estão alinhados ou contradizem o resultado de Mao. Portanto, foi examinada mais de perto a distribuição do número de hiperespecialistas por desafio, que é apresentado na Tabela - 4.4. Na tabela, é possível observar que 43% de todos os desafios não contam com a presença de hiperespecialistas. Desconsiderando os desafios que não têm hiperespecialistas, 62% dos desafios verificados têm apenas um hiperespecialista, o que, de certo modo, está de acordo com a observação de Mao e seus colegas.

**Tabela 4.4:** Número de hiperespecialista por desafio

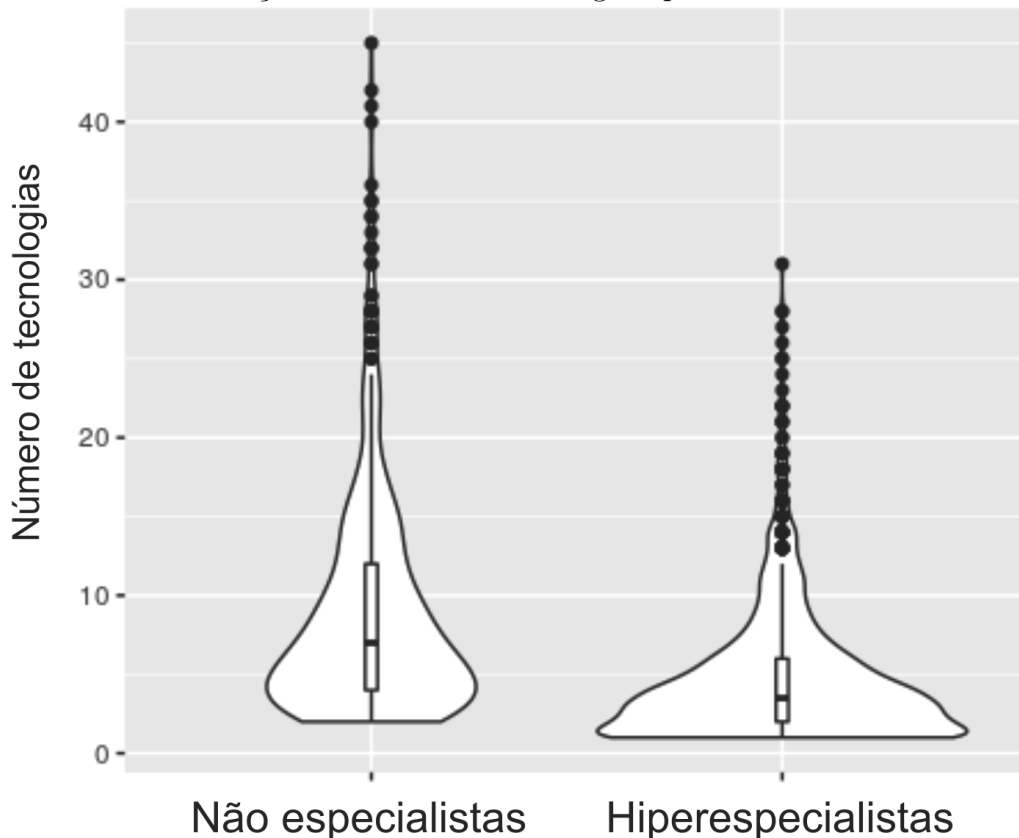
Desafios	Hiperespecialistas
8171	0
6525	1
2272	2
865	3
826	$\geq 4$

**QP1.6 É possível diferenciar hiperespecialistas e não especialistas com base em dados de participação de desafios?**

Para responder a essa questão, analisou-se a diferença entre os hiperespecialistas e não especialistas em relação ao número de tecnologias a que os usuários estavam em contato, o número de desafios em que os usuários participaram, o número de vitórias e a proporção de vitórias por desafio. A primeira característica analisada foi a quantidade total de diferentes tecnologias em que os usuários tiveram contato. Esse número foi obtido por meio da análise das tecnologias necessárias nos desafios que os usuários participaram. A distribuição para os hiperespecialistas e não especialistas é mostrada na Figura - 4.7. É possível observar que a maior parte dos hiperespecialistas e não

especialistas está na parte inferior do gráfico. Ao analisar as medianas e os *whiskers*, é possível verificar que os hiperespecialistas tiveram contato com um número menor de tecnologias diferentes em relação aos não especialistas. Isso é esperado, já que os hiperespecialistas buscam desafios que exigem uma (ou algumas) tecnologias específicas, enquanto os não especialistas contribuem em um número maior de desafios, dando menos atenção às tecnologias que contribuem para desafios mais simples. Tal diferença aparente foi confirmada pelo resultado do teste de Mann-Whitney-Wilcoxon, ( $p\text{-value} < 0,001$ ), indicando que o número de tecnologias de cada usuário em ambas as distribuições não é igual. O cliff-delta ( $cliff\text{-delta}=0,379$ ) mostra que há um tamanho de efeito médio, maior para os não especialistas, confirmando o que é observado nos gráficos.

**Figura 4.7:** Distribuição de diferentes tecnologias que os usuários tiveram contato.

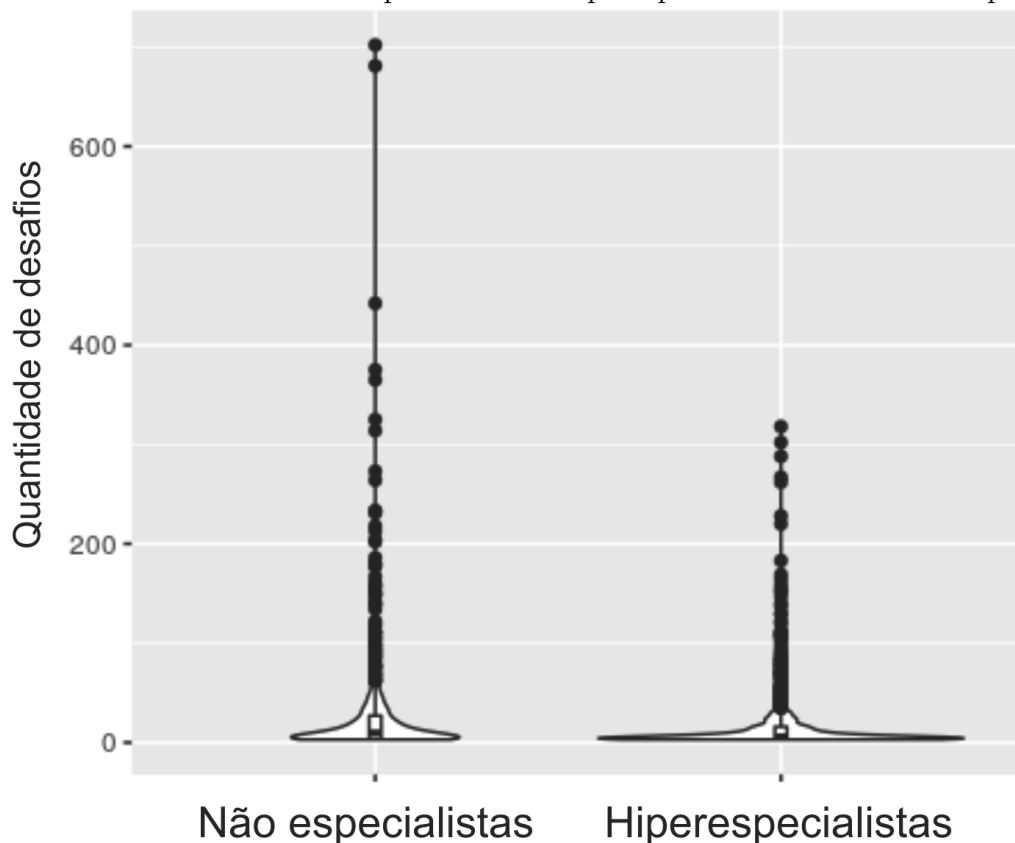


A Figura - 4.8 apresenta a distribuição do número de desafios de que os hiperespecialistas e não especialistas participaram. Novamente, foi utilizado o teste estatístico de Mann-Whitney-Wilcoxon para comparar as duas distribuições. Verificou-se que o número de desafios em ambas as distribuições não é igual ( $p\text{-value} < 0,001$ ). Mais uma vez foi encontrado um tamanho de efeito pequeno ( $cliff\text{-delta}=0,162$ ), indicando valores

ligeiramente superiores para não especialistas. Uma possível explicação para esse resultado é que os não especialistas se concentram em uma variedade maior de desafios, uma vez que não são restritos por um conjunto de tecnologias.

Ao analisar a participação dos hiperespecialistas em desafios, observou-se que 60% dos hiperespecialistas participaram apenas de desafios que exigem uma tecnologia em que são especialistas. Além disso, 83% desses hiperespecialistas participaram de menos de 10 desafios (conforme representado pelo achatamento do gráfico). Para os não especialistas, 48% (408 usuários) participaram de menos de 10 desafios, e que existem 5 *outliers* que submeteram mais de 350 desafios, com um máximo de 702(!). Analisou-se o subconjunto de desenvolvedores que participaram de mais de 100 desafios e foram identificados 45 não especialistas (5,4%) e apenas 29 hiperespecialistas (2,3%). Este fato é outra evidência de que os não especialistas participam de mais desafios.

**Figura 4.8:** Número de desafios por usuário: hiperespecialistas versus não especialistas



O número de vitórias é outra característica utilizada para comparar hiperespecialistas e não especialistas. Na Figura - 4.9, é possível observar muitos usuários classificados como hiperespecialistas sem vitórias. Esse fenômeno ocorre em menor grau nos usuários

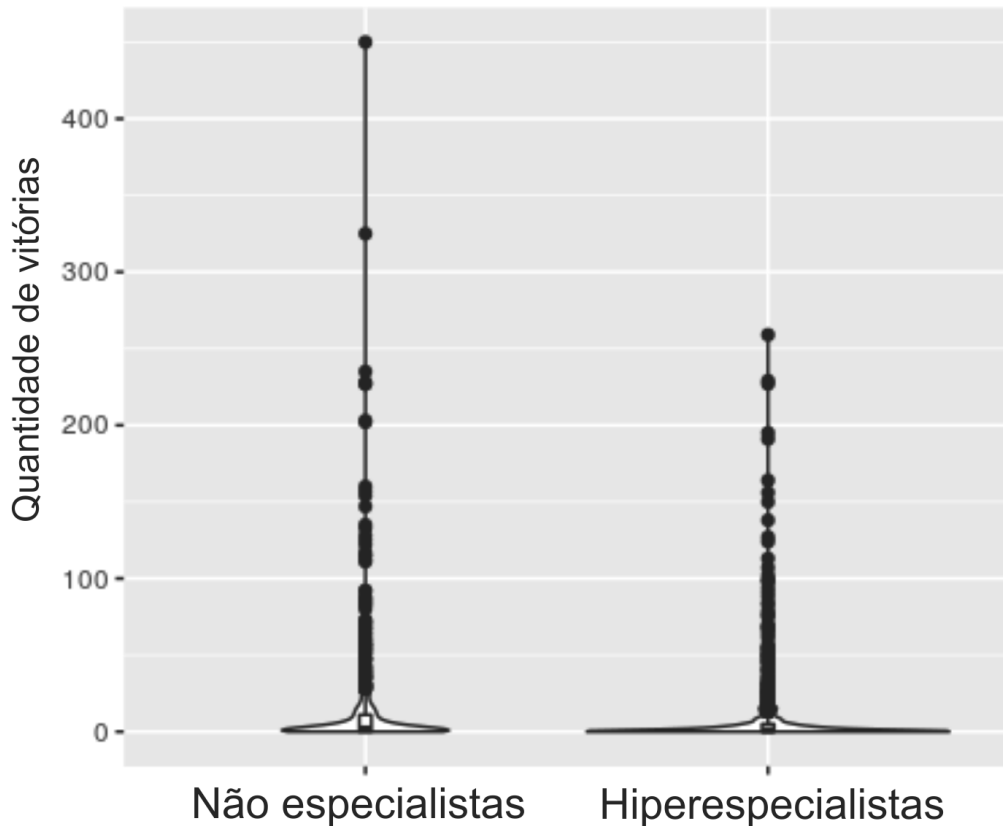
que não foram catalogados como hiperespecialistas, pois a distribuição no gráfico é mais homogênea em relação aos hiperespecialistas. Como em observações anteriores, o teste de Mann-Whitney-Wilcoxon também indicou uma diferença significativa entre as vitórias e os grupos ( $p\text{-value} < 0,001$ ). Um pequeno tamanho de efeito ( $cliff\text{-}delta = 0,168$ ) foi encontrado em favor dos não especialistas.

Analisando o grupo de hiperespecialistas, foi verificado que existem 523 (41,6%) usuários que não possuem vitórias. Quanto aos não especialistas, foi verificado que 236 (28,4%) usuários não possuem vitórias. Não foi encontrada nenhuma explicação para esse fato. Para aqueles que venceram pelo menos um desafio, 93% (683 hiperespecialistas) têm mais de 70% de suas vitórias nos desafios que exigem tecnologias que os usuários eram hiperespecialistas.

Outro fato interessante é que poucos membros são responsáveis pela maior parte das vitórias. Analisando os 5% do topo para cada grupo, foi descoberto que, para os hiperespecialistas, 63 usuários são responsáveis por 53,8% das vitórias do grupo (5227 de 9715); para não especialistas, os números são semelhantes: 46 pessoas foram responsáveis por 50,98% das vitórias do grupo (5721 de 11222). Analisando os 20% superiores, notou-se que a regra 80/20 é observada (no geral, 20% da população é responsável por 84,4% das vitórias).

É possível observar que o número dos não especialistas é maior do que o número dos especialistas, tanto em termos de desafios quanto de vitórias. Testando a correlação dessas duas distribuições, foi verificado que elas são, de fato, fortemente correlacionadas (0,788) considerando todos os usuários; 0,746 considerando apenas os hiperespecialistas; 0,832 considerando não especialistas. Portanto, fica evidente que quanto mais usuários competem, mais eles ganham, independentemente de sua especialização. Assim, para fazer uma comparação justa, decidiu-se verificar se a proporção de vitórias ( $\#vitórias/\#desafios$ ) é diferente quando comparamos hiperespecialistas e não especialistas. A distribuição dessa razão é mostrada na Figura - 4.10. É perceptível que a base da distribuição dos hiperespecialistas ainda é maior se comparada à base do gráfico dos usuários não especialistas (já que existem mais hiperespecialistas sem vitórias). No entanto, as distribuições agora são mais claras (já que não existem *outliers*, porque dados foram normalizados). Calculando novamente o teste de Mann-Whitney-Wilcoxon, o resultado mostra diferença entre as populações ( $p\text{-value} < 0,001$ ). No entanto, o tamanho de efeito é insignificante ( $cliff\text{-}delta = 0,1163$ ) em favor dos não especialistas.

**Figura 4.9:** Número de vitórias por usuário: hiperespecialistas vs. não especialistas

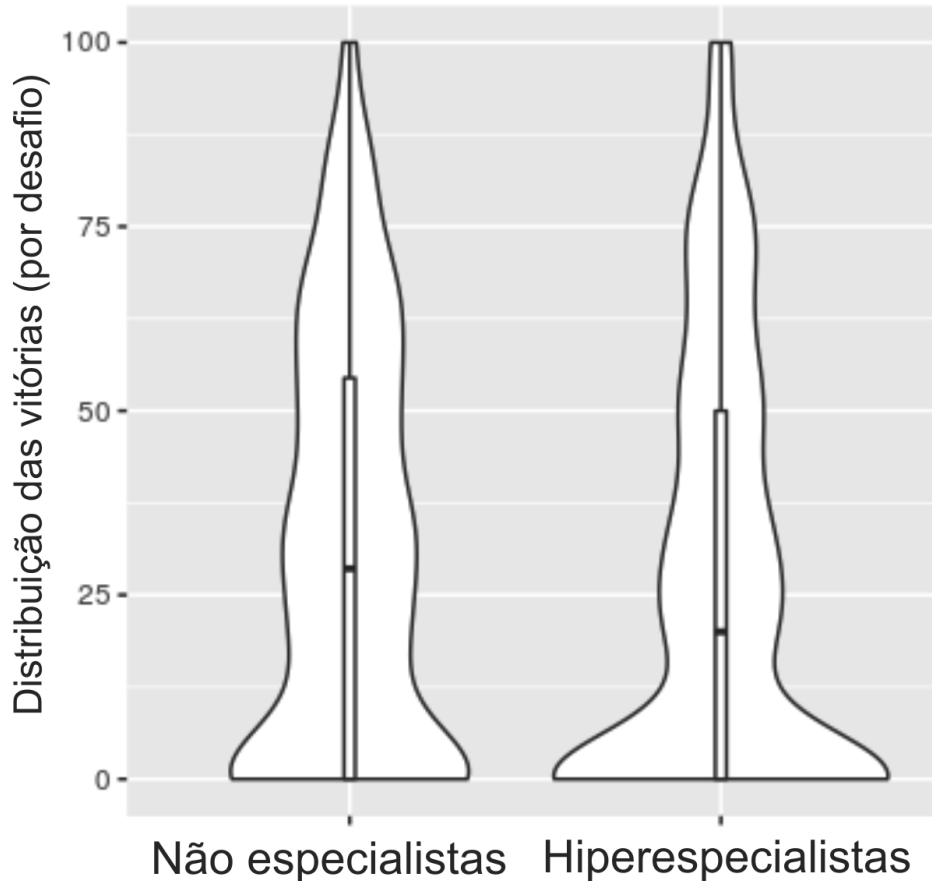


### 4.3 Implicações

Esta fase tem implicações diretas em pelo menos três dos quatro pilares do *crowdsourcing* (empresas, plataformas e desenvolvedores).

**Empresas.** É possível observar uma grande parcela de desenvolvedores da plataforma Topcoder que foram classificados como especialistas em uma ou mais tecnologias (QP1.3). Assim, as empresas podem ter isso em mente ao disponibilizar suas tarefas. Contando com um grupo de especialistas, pode ser fácil encontrar ajuda para resolver problemas específicos e receber rapidamente respostas de desenvolvedores experientes. Também foi possível observar que há muitos não especialistas que fornecem artefatos vitoriosos. Portanto, as empresas também podem contar com esses usuários para apoiar diferentes tipos de situações. Especificamente para o Topcoder, foram categorizados os desafios, tecnologias e desenvolvedores, mostrando as tecnologias mais solicitadas e a população de hiperespecialistas por tecnologia.

**Figura 4.10:** Relação de vitória (vitória/desafio): comparando hiperespecialistas e não especialistas



**Desenvolvedores.** Acredita-se que os resultados obtidos podem incentivar os desenvolvedores a participar e mantê-los nas plataformas de *crowdsourcing*. Como foi possível notar ao responder QP1.6, embora exista um alto número de hiperespecialistas, o número de vitórias alcançado por usuários mais generalistas é maior do que pelos hiperespecialistas. Assim, há oportunidade para diferentes perfis, e é possível ganhar dinheiro contribuindo para projetos de *crowdsourcing* em ambos os casos.

**Plataforma.** A plataforma é provavelmente a que mais se beneficia com os resultados da pesquisa. Em primeiro lugar, ficou evidenciado que existe um grande conjunto de tecnologias sem especialistas identificados (QP1.3). A plataforma poderia investir em métodos para ajudar e incentivar os desenvolvedores a se especializarem nessas tecnologias, além de incentivar os clientes a criar demandas nessas tecnologias. Em segundo lugar, há poucos usuários que efetivamente submetem nos desafios, pois só foi possível analisar 2.086 usuários que enviaram soluções para pelo menos três desafios. Em terceiro lugar, na



Etapa 1, ficou evidente que 66% da amostra abandonou a plataforma antes dos primeiros 6 meses. Portanto, os mantenedores da plataforma poderiam propor estratégias para manter os novos membros engajados, reduzindo assim essa alta taxa. Quarto, foi demonstrado que existe uma “classe de elite” que responde pela maioria das vitórias (QP1.6). É possível que, seguindo as observações citadas neste parágrafo, usuários mais habilidosos possam se engajar na plataforma e ter chances de vencer mais os desafios. Finalmente, identificando os hiperespecialistas, é possível para a plataforma usar essa caracterização para explorar as peculiaridades de cada grupo a fim de melhorar seu modelo de negócios, ou melhor atender às demandas, atraindo e mantendo mais desenvolvedores que resultarão em melhores softwares e, conseqüentemente, satisfação do cliente.

**Pesquisadores.** Este trabalho pode inspirar replicações, por exemplo, para analisar a manifestação da hiperespecialização em outros modelos de *crowdsourcing* em engenharia de software e outros domínios. Acredita-se que o modelo de micro tarefa é mais vantajoso para os hiperespecialistas. Por fim, é esperado que este trabalho auxilie na evolução da pesquisa relacionada aos perfis de contribuição de usuários de *crowdsourcing* para desenvolvimento de software, além de inspirar futuras investigações sobre esse fenômeno.

## 4.4 Limitações e Ameaças à Validade

O principal ponto questionável deste trabalho é a definição do hiperespecialista. Não foram encontrados argumentos na literatura para apoiar as escolhas feitas. Assim, o limite de contribuição de 75% em uma única tecnologia foi definido pelo fato de que poderia haver várias tecnologias em cada um dos desafios analisados. Um limite superior ou inferior pode não retratar claramente as respostas às perguntas feitas.

Outra ameaça potencial diz respeito à decisão de classificar as tecnologias em categorias genéricas. Esta ação foi tomada porque o objetivo deste trabalho é caracterizar o perfil dos hiperespecialistas em uma determinada tecnologia. Seria desafiador, ou mesmo impossível, identificar esse perfil considerando tecnologias muito refinadas (*frameworks*, componentes), como versões de uma tecnologia específica (.NET 2.0/.NET 3.0, Oracle 9i/Oracle 10g). Ao generalizar as tecnologias, acredita-se que foi possível fornecer um bom ponto de vista sobre este fenômeno.

O limite dos resultados deste trabalho está relacionado à plataforma e à amostra. Foram analisados tarefas e desenvolvedores da plataforma Topcoder, uma plataforma que implementa um modelo de *crowdsourcing* baseado em concorrência. Além disso, na segunda etapa, o foco foi nos desafios de desenvolvimento que ofereciam recompensas financeiras para os vencedores, sem analisar os subtipos dos desafios. Os resultados não

se aplicam a outras plataformas ou modelos. Além disso, foram analisados usuários que enviaram soluções para pelo menos três desafios do tipo desenvolvimento. Este limiar foi definido com base na distribuição dos envios, optando por analisar o terceiro quantil (top 25%) dos usuários.

## 4.5 Conclusão

Nesta fase, foi possível evidenciar, de duas formas distintas, a existência e as características dos usuários que focam seus esforços em tipos específicos de tarefas ou tecnologias dentro de plataformas de *crowdsourcing* de desenvolvimento de software.

Os resultados da Etapa 1 mostraram que a hiperespecialização pode ser evidenciada em termos do tipo de tarefa escolhida pelos usuários. Verificou-se que, entre aqueles que continuaram contribuindo por 18 meses após sua estreia, 94% contribuem para tarefas do mesmo tipo (desenvolvimento, projeto ou Ciência de Dados). Além disso, foi evidenciado uma alta taxa de abandono (66% da amostra contribuíram apenas no primeiro período de análise). É importante para a plataforma reverter essa situação a fim de manter um grupo maior de trabalhadores que podem construir sistemas de maior qualidade.

Os resultados da Etapa 2 mostraram que, em desafios de desenvolvimento, 60% dos usuários são classificados como hiperespecialistas. Existem hiperespecialistas em 45% das categorias de tecnologia, no entanto, tecnologias famosas e consolidadas como Docker, MySQL e PostgreSQL não contam com especialistas. Ao comparar especialistas e não especialistas, foi verificado que os não especialistas submetem para desafios com uma ampla gama de tecnologias, participam de mais desafios e, surpreendentemente, apresentam uma taxa de vitórias maior do que os hiperespecialistas.

---

## Perfil de contribuição

---

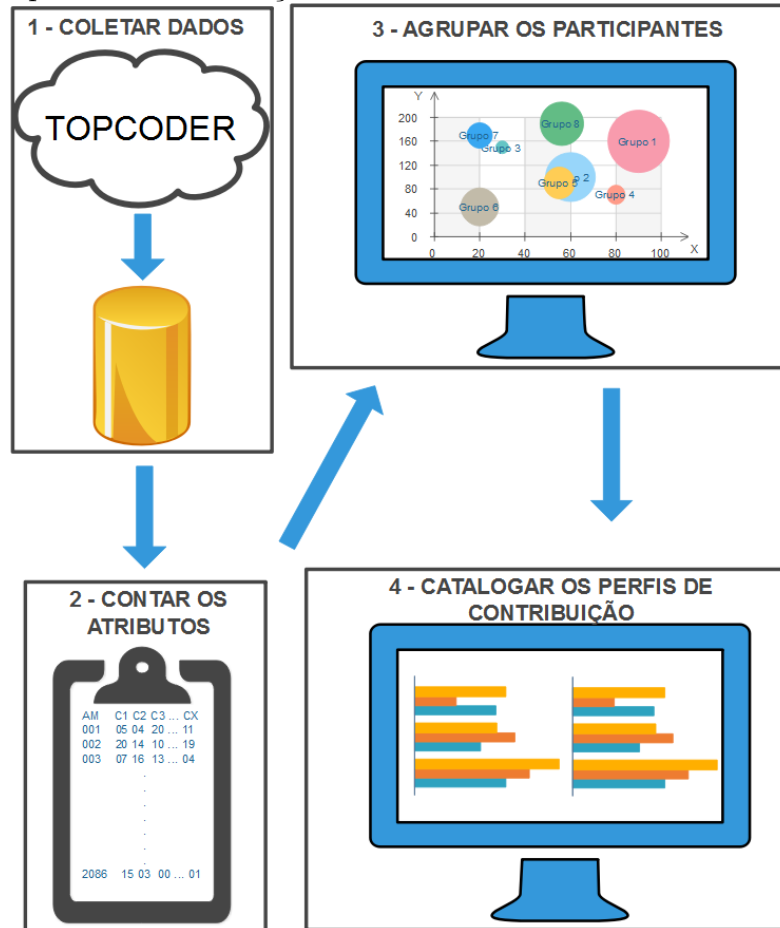
Este capítulo reporta-se aos detalhes referentes à análise de diferentes perfis de contribuição bem como seus resultados, com o intuito de responder à questão de pesquisa dois (QP2).

### 5.1 Método de Pesquisa

O método utilizado para a identificação do perfil de contribuição dos usuários é apresentado na Figura - 5.1. Similar ao que foi definido nas fases da análise da hiperespecialização, o passo inicial deste estudo é baseado na coleta de dados (os detalhes sobre a coleta dos dados estão descritos na Seção 3.3).

No passo 2 da figura, foram escolhidas informações que estão disponíveis no banco de dados, e que foram consideradas importantes para a identificação de perfis dentro da plataforma. Essas características levam em consideração dados como total de conquistas e de participações, entre outras informações detalhadas na Seção 5.2. Algumas delas foram totalizadas com base na quantidade de vezes que a ação foi executada. Por exemplo, o total de vitórias representa quantas vezes o participante recebeu recompensa por vencer desafios, não necessariamente por ficar em primeiro lugar. Outros atributos foram representados pelo total em dias que os usuários necessitaram para completar uma ação. Por exemplo, quantos dias determinado usuário precisou para fazer a primeira submissão em um desafio. Todos os usuários tiveram cada uma das características contabilizadas e armazenadas em um vetor. Deste modo, cada linha da matriz representa o perfil de contribuição de um dos 2.086 usuários da amostra.

**Figura 5.1:** Representação gráfica do método utilizado na fase de análise da constituição do perfil de contribuição



O passo 3 do método é o agrupamento dos participantes baseado nos atributos. Para isso, foi utilizado um algoritmo de agrupamento denominado *Model-based Clustering* (Fraley e Raftery, 2002; Fraley et al., 2012). Esse algoritmo foi utilizado neste trabalho, devido ao fato de o mesmo não necessitar de um conjunto de parâmetros de agrupamento inicial, utilizando apenas os dados dos usuários para gerar os agrupamentos. Além disso, esse algoritmo aceita como entrada de dados uma matriz, que é exatamente a estrutura de dados que armazena as informações contabilizadas no passo anterior. Desse modo, este algoritmo consegue identificar e separar os usuários que possuem perfis semelhantes, mesmo com os dados em múltiplas dimensões. Gerando, como resultado, agrupamentos ou *clusters* com os usuários que possuem maior afinidade em seu perfil de contribuição na plataforma.

Após definidos os grupos de participantes, foi necessário analisar cada um dos grupos a fim de identificar suas principais características. Para isso, foi novamente utilizada

estatística descritiva suportada por gráficos. Desse modo, manualmente, as principais características de cada grupo foram identificadas.

## 5.2 Filtragem e análise dos dados

Os usuários selecionados para esta fase são os mesmos usuários filtrados na segunda etapa da fase da hiperespecialização. São 2.086 usuários que possuem ao menos três submissões em desafios distintos. Esses desafios são do tipo desenvolvimento, devem ter ao menos uma tecnologia relacionada com o desafio e oferecer uma recompensa em dinheiro para quem vence o desafio. As submissões contam apenas quando o usuário está competindo em busca da recompensa por terminar o desenvolvimento do artefato de software requerido.

Para cada um desses usuários foram contabilizados atributos que serviram de base para definir o perfil de cada usuário. As características escolhidas e contabilizadas são:

- Total de dias na comunidade: diferença em dias entre data de início da coleta e a data do cadastro do usuário;
- Total de dias para a primeira submissão: diferença em dias entre a data do cadastro do usuário e a data do primeiro desafio que o usuário submeteu;
- Total de dias em competição: diferença em dias entre o último desafio com submissão e o primeiro;
- Total de conquistas: total de metas atingidas pelos usuários. Por exemplo, primeira vitória, 25 vitórias, primeira postagem no fórum entre outros;
- Total de desafios com submissão: quantidade de desafios de desenvolvimento que o usuário submeteu alguma solução;
- Total de vitórias: total de vezes que o usuário ficou em uma colocação nos desafios que o recompensava;
- Total de tecnologias: quantidade de tecnologias distintas presentes nos desafios em que os usuários concorreram (definido na segunda etapa da fase de análise da hiperespecialização)
- Total de habilidades: total de itens que representam as diferentes habilidades que o usuário julga possuir;

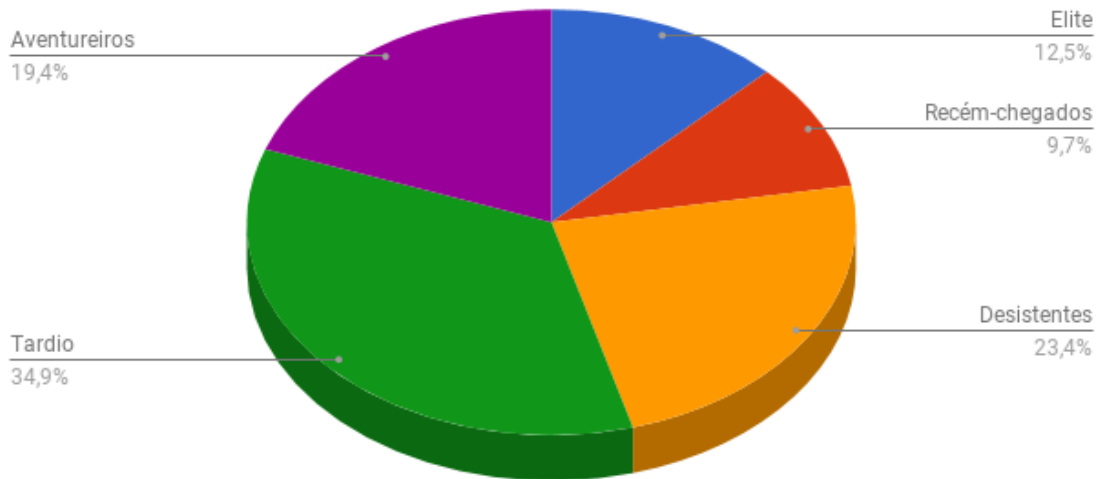
- Hiperespecialização: ação dos usuários conhecerem a fundo poucos assuntos (definido na segunda etapa da fase de análise da hiperespecialização).

É importante destacar que o atributo total de tecnologias foi concebido pela junção de tecnologias semelhantes, assim como descrito na Seção 4.2.2. Desse modo, o total de habilidades, apesar de ser arbitrário (definido pelo usuário), foi utilizado com o intuito de complementar o atributo total de tecnologias, ou seja, o termo “habilidade” faz menção aos conhecimentos que os usuários julgam possuir, como conhecimento em manipulação de *Strings*, conhecimento de grafos, ordenação, expressões regulares e tecnologias em geral. O termo especialidade definido na fase 2 do estudo da hiperespecialização e utilizado nesta etapa visa demonstrar que os usuários da plataforma buscam, em geral, um conjunto específico de condições para participarem dos desafios, buscando assim, mostrar que eles são especialistas em tais critérios. No caso deste estudo foi utilizado principalmente as tecnologias presentes nos desafios que os usuários concorreram enviando suas respostas para representar a especialização dos usuários.

### 5.3 Resultados

Após a contagem dos atributos para todos os usuários da população amostral, o algoritmo de clusterização foi executado, tendo como entrada os resultados das contagens. A execução do *Model-based Clustering* resultou em uma divisão de cinco grupos que estão dispostos segundo a Figura - 5.2. Os grupos foram denominados como: elite (261 usuários), recém-chegados (203 usuários), desistentes (489 usuários), tardios (728 usuários) e usuários aventureiros (405 usuários).

A Tabela - 5.1 mostra o centro dos atributos para cada um dos cinco grupos encontrados. Esses dados foram utilizados para caracterizar os grupos e serão comentados posteriormente com as definições de cada grupo. Com o intuito de melhorar a visualização dos dados na tabela, as características foram substituídas pela ordem apresentada na Seção 5.2. Por exemplo, a primeira característica apresentada é o “Total de dias na comunidade” que foi substituído por “C1”; a segunda característica foi “Total de dias para a primeira submissão”, que foi substituída por “C2”; e assim para todos os outros atributos. A Tabela - 5.2 representa a contração de todos os atributos. O único atributo que não está representado nas tabelas é o da hiperespecialização devido ao fato de o mesmo ser zero quando o usuário não é hiperespecialista e um quando o usuário é hiperespecialista. Informações complementares como mínimo, máximo, primeiro quartil, terceiro quartil e média dos grupos sobre cada um dos agrupamentos estão disponíveis no Apêndice A.

**Figura 5.2:** Distribuição dos usuários entre os grupos identificados**Tabela 5.1:** Centro das característica nos grupos

Grupos	C1	C2	C3	C4	C5	C6	C7	C8
Grupo 1	2.781,90	358,07	1.871,62	18,98	88,76	53,65	15,62	30,06
Grupo 2	650,07	45,71	40,19	2,82	3,82	0,76	3,28	5,07
Grupo 3	3.047,05	553,84	196,55	7,87	4,80	0,41	2,69	10,14
Grupo 4	2.879,90	749,11	861,34	10,20	18,50	7,50	7,00	15,77
Grupo 5	2.072,40	12,25	272,98	3,58	6,84	1,88	3,81	4,86

### 5.3.1 Grupo 1: Elite

O primeiro grupo encontrado é formado por 261 usuários, tidos como “elite”. Estes possuem a mais alta taxa de vitórias dentre os grupos identificados nesta fase (53,65 vitórias). Contudo, essa não é a única característica importante. Eles possuem o maior número de dias em participação na plataforma (1.871,62 dias), bem como tiveram contato com a maior diversidade de tecnologias presente em desafios (15,62 tecnologias) e possuem o maior número de habilidades (30,06 habilidades). Um grupo com alta taxa de vitórias e grande quantidade de tecnologias e habilidades está relacionado aos resultados encontrados na Etapa 2 da análise da hiperespecialização, que mostrou que os usuários que trabalham com mais tecnologias vencem mais. Essas características refletem diretamente na quantidade de desafios que os usuários desse grupo submetem, visto que, é detentor do maior número de desafios com, ao menos, uma submissão (88,76 desafios, na média).

**Tabela 5.2:** Relação das abreviaturas com os nomes completos das características

Abreviação	Nome completo
C1	Total de dias na comunidade
C2	Total de dias para a primeira submissão
C3	Total de dias em competição
C4	Total de conquistas
C5	Total de desafios com submissão
C6	Total de vitórias
C7	Total de tecnologias
C8	Total de habilidades

Outro atributo relevante deste grupo é a grande presença de copilotos. Esse atributo não foi utilizado para o algoritmo de agrupamento, contudo, verificou-se que, dos 261 usuários, 58 são copilotos, sendo que, em toda a população, existem apenas 98 copilotos, ou seja, 59% dos copilotos estão presentes neste grupo.

Um grupo composto por usuários acima da média em participações e vitórias, formando uma espécie de elite, não é uma exclusividade do *crowdsourcing*. Isto também foi observado em outras comunidades *online* como na área de Q&A (Furtado et al., 2013) ou na Wikipédia (Kittur et al., 2007). O trabalho de recomendação de participantes relacionados ao *crowdsourcing* (Mao et al., 2015b) também indica ser possível a existência dessa elite de membros, visto que é baixa a diversidade de usuários selecionados para completar as tarefas.

Este é um grupo importante para o ecossistema, visto que é constituído por grande parte dos usuários que vencem os desafios propostos pelos clientes, bem como, da maior parte de usuários que auxiliam as empresas na criação e manutenção dos desafios, os copilotos.

### 5.3.2 Grupo 2: Recém-chegados

O grupo 2 é o menor sendo formado por 203 usuários. Sua denominação é “usuários recém-chegados” ou “usuários novatos”. São os que possuem a menor taxa de quantidade de dias na plataforma (650,07 dias). Assim como os usuários aventureiros, os novatos submetem rapidamente em seus primeiros desafios (45,71 dias), em comparação com os grupos tardios e elite. Contudo, a taxa de vitórias é pequena (0,76 vitórias). Uma característica que pode indicar baixa interação com a plataforma é o total de conquistas (2,82 conquistas), sendo o grupo com menor taxa de conquistas.

A principal hipótese para essa ação é a desmotivação em não conseguir vencer. A dificuldade em vencer pode ser abstraída em duas hipóteses: (i) os usuários não possuem



o conhecimento necessário para completar as tarefas e desistem. Essa hipótese é reforçada por um atributo que não foi utilizado para a clusterização, a quantidade média de desafios que os usuários se inscrevem. Para o grupo dos novatos, o valor médio é de 13,18 desafios inscritos, contudo eles só conseguem submeter em média 3,84 desafios; (ii) os usuários enfrentam dificuldades na plataforma, seja para encontrar as tarefas recomendadas, seja para entender e concluir a tarefa.

Assim como reportado por (Zanatta et al., 2017), os novos usuários enfrentam dificuldades em participar da plataforma, e em muitos casos acabam encerrando sua participação. Desse modo, este grupo merece atenção, justamente pela importância de existir a renovação dos desenvolvedores presentes na comunidade, bem como o crescimento e a diversificação de experiências que são pontos essenciais para a criação de softwares de qualidade por meio do *crowdsourcing*.

### 5.3.3 Grupo 3: Desistentes

O grupo 3 conta com 489 usuários, segundo maior grupo, sendo catalogado como “usuários desistentes”. Essa classificação foi proposta justamente pelo fato de este grupo ter o segundo menor tempo em competição (196,55 dias). Este é um grupo que possui a pior taxa de vitórias (0,41 vitórias) dentre os grupos encontrados e possui também a menor variância de tecnologias participadas nos desafios (2,69 tecnologias). Outra importante característica deste grupo é a demora em começar a competir (553,84 dias), sendo a segunda maior entre os grupos identificados.

As hipóteses relacionadas à desistência dos usuários desse grupo são as mesmas apresentadas para o grupo dos novatos. A falta de conhecimento para conquistar as vitórias e os problemas relacionados à plataforma podem desestimular os usuários a participarem levando-os a começar a contribuir tardiamente bem como desistirem de competir rapidamente.

A análise perfil de usuários em ambientes de *crowdsourcing* não é uma exclusividade deste trabalho, sendo possível encontrar literatura relacionada em outras comunidades online. No trabalho de Thelwall (2008) foi verificado que cerca de um terço dos usuários da rede social MySpace acaba desistindo de participar prematuramente. Saremi e Yang (2015) citam que a carreira profissional e a situação dos usuários na plataforma podem influenciar a decisão do usuário em desistir de contribuir com a plataforma.

Hipotetiza-se que esse é um grupo de usuários que dificilmente voltaria a competir na plataforma, devido à experiência inicial. Ações no intuito de fazê-los não desistir de competir podem ser analisadas. Por exemplo, nivelar os desafios mostrando quais são

mais indicados para os novatos, intensificar a distribuição de conhecimento e incentivar a ajuda mútua são ações que podem ser tomadas para tentar mantê-los competindo.

### 5.3.4 Grupo 4: Tardios

O grupo 4 é composto por 728 usuários classificados como “tardios”. É um grupo de usuários que, em geral, demorara muito tempo para efetivamente começar a submeter seus artefatos de software, na média 749 dias. Apesar dessa demora excessiva, eles possuem o segundo maior nível médio de submissão nos desafios (18,50) e de vitórias (7,50) entre os grupos encontrados. Outra característica importante deste grupo é que, em geral, apesar da demora em começar a competir, eles se mantêm ativos por bastante tempo (861,34 dias), visto que seu tempo médio de competição também é o segundo maior. Além disso, é notável o valor da diversidade de tecnologias (sete tecnologias) e o total de habilidades (15,77) que usuários participam. Isso indica a variedade de assuntos em que os usuários contribuem.

É possível traçar algumas hipóteses para tentar entender qual é a causa desta grande demora para começar a competir:

- Os usuários se cadastraram, mas por dificuldades como, falta de conhecimento, problemas para encontrar as tarefas mais indicadas ou outros problemas relacionados com a plataforma, eles desistiram de participar. Contudo, em um segundo momento, eles vencem essas barreiras e começam a competir, tornando-se usuários de valor. Um atributo não utilizado para a clusterização, mas que pode reforçar esta hipótese é a alta média de dias que os usuários demoraram para se inscrever (sem necessitar submeter) no primeiro desafio, que é de 518,94 dias.
- Também é plausível pensar que os usuários destes grupos estavam competindo em outras categorias, visto que um dos resultados da primeira etapa da análise da hiperespecialização indica que muitos usuários participam em outras categorias de desafios como a Ciência de Dados.

Analisando hipótese de que os usuários poderiam estar competindo em outras categorias de desafios, foi verificado que, 71,1% dos usuários deste grupo (518 usuários) possuem algum tipo de participação em desafios classificados como *design*, SRM ou *marathon*, independentemente do papel desempenhado na participação do desafio. Os 28,9% restantes são compostos por 210 usuários que não possuem nenhuma interação com outros desafios. Analisando os subgrupos, foi verificado que os usuários que possuem interação com outros tipos de desafio demoram, em média, 867,13 dias para começar a

submeter, já os que não possuem interação com outros desafios demoram 468,88 dias. Devido à diferença entre os resultados, é possível observar que o fato dos usuários participarem de outros desafios impacta no tempo pra iniciar a competir nos desafios de desenvolvimento.

Analisou-se então, se os usuários que possuíam participações em outros desafios começaram a participar destes outros desafios antes ou depois de submeterem nos desafios de desenvolvimento. Dos 518 usuários que possuíam alguma interação com outros desafios, 72,8% (377 usuários) o fizeram antes da submissão nos desafios de desenvolvimento; 25,9% (134 usuários) interagiram somente após a primeira submissão; e 1,3% (sete usuários) tiveram interações registradas no mesmo dia. Essa análise resultou em indícios que auxiliam na compreensão da demora que os usuários apresentaram para começar a submeter, contudo, novas análises podem confirmar tal resultado. É importante ressaltar que a seleção destes desafios leva em consideração apenas se o usuário se inscreveu para participar do desafio, sem necessariamente o usuário ter submetido. Desse modo, tal análise pode ser injusta. Contudo, o resultado da mesma pode ser um indício para que novas análises investiguem os resultados com mais cautela.

O fato de alguns usuários participarem de outros tipos de desafios é compreensível, principalmente pelo ao fato de as competições denominadas SRMs servirem justamente para que os usuários possam testar suas habilidades com desafio propostos por usuários mais experientes da plataforma<sup>1</sup>.

Este grupo merece atenção, pois, se for possível diminuir o tempo de adaptação destes usuários, será possível contar com novos bons desenvolvedores com um menor investimento de tempo. Removendo essas dificuldades iniciais, é possível que o tempo de permanência na plataforma aumente, gerando vantagens para todo o ecossistema.

### 5.3.5 Grupo 5: Aventureiros

Por fim, o grupo 5 é constituído por 405 usuários com a denominação de usuários “aventureiros”. Sua classificação é proposta a partir das hipóteses apontadas para o entendimento de sua participação na plataforma. Devido ao fato de este não ser um grupo que possui uma característica que destoa das outras, é um grupo que fica com seus atributos, em geral, na média dos outros grupos.

Uma das características deste grupo que difere dos outros grupos é o total de dias para a primeira submissão. Os usuários deste grupo começam a submeter rapidamente (em média em 12,25 dias), característica essa que também contraria as ações dos grupos

---

<sup>1</sup><https://www.topcoder.com/community/competitive-programming/>

tardios e elite. Um atributo que não foi utilizado no agrupamento dos usuários, mas que pode ajudar a entender este grupo é a média de dias que os usuários necessitam para se inscreverem no primeiro desafio, sem necessariamente submeter. Para este grupo, a média é de apenas 5,8 dias, ou seja, em geral, os usuários se cadastram rapidamente e enviam suas contribuições rapidamente, porém, vencem poucos desafios (1,88 desafios com vitórias).

O conjunto de características de se inscrever e submeter rapidamente, vencer pouco e parar de contribuir rapidamente pode indicar, ao menos, três situações: (i) os usuários que vencem atingem seus objetivos e param de competir; (ii) os usuários que não vencem perdem sua motivação inicial e desvencilham-se da plataforma; (iii) estão competindo em outros tipos de desafios.

Assim como feito no grupos dos usuários tardios, foram analisadas as participações dos usuários do grupo em outros tipos de desafios. Verificou-se que 25,7% (104 usuários) participaram de outro tipo de desafio; e que 74,3% não o fizeram (301 usuários). As participações em outros desafios pouco influenciaram na média de tempo para a primeira submissão, sendo de 14,63 dias para os usuários que participaram de outros desafios e 11,37 para os que não participaram. Além disso, nota-se que apenas 19% dos usuários que participaram de outro desafio (20 usuários) interagiram com algum desafio após a última submissão nos desafios de desenvolvimento. Esses resultados indicam que uma pequena parcela do grupo (4,9% ou 20 de 405 usuários) começa a participar de outros tipos de desafios após parar de contribuir em desafios de desenvolvimento. O restante da população que participa de outros tipos de desafios o faz antes ou intercalado com os desafios de desenvolvimento. Novamente é importante destacar que esta análise pode não levar em conta submissões (como feito para os desafios de desenvolvimento). Contudo, ela pode indicar se parte dos usuários migram dos desafios de desenvolvimento para outros tipos de desafios.

Uma característica que pode indicar que os usuários deste grupo possuem um objetivo de curto ou médio prazo é o baixo número médio de conquistas, uma taxa de 3,58 conquistas. Observando as 10 conquistas que os usuários deste grupo mais atingiram, Tabela - 5.3 é possível identificar que a maior parte das conquistas é superficiais. Seis delas indicam a primeira vez que o usuário conseguiu algo como a primeira submissão ou a primeira postagem no fórum, 3 delas indicam que os usuários são participantes de desafios e apenas uma indica interação pouco mais intensa com a plataforma (100 postagens nos fóruns).

Outro ponto que pode indicar baixo nível de interação com a plataforma é o número total de habilidades (4,86 habilidades), menor entre todos os grupos. O valor máximo para

**Tabela 5.3:** As dez conquistas que os usuários do grupo 5 mais atingiram

Conquista	Total usuários
<i>First Passing Submission</i>	380
<i>First Forum Post</i>	366
<i>First Placement</i>	226
<i>First Win</i>	178
<i>iOS Developer Challenger Badge</i>	62
<i>One Hundred Forum Posts</i>	50
<i>iOS Participant Badge</i>	34
<i>First Rated Algorithm Competition</i>	33
<i>Participant Badge</i>	29
<i>First Marathon Competition</i>	20

esse atributo também é o menor dentre os grupos (16 habilidades). Como esse atributo é informado manualmente pelos usuários, é possível inferir que não é preocupação dos usuários deste grupo ficar interagindo com outros afazeres, além de competir nos desafios.

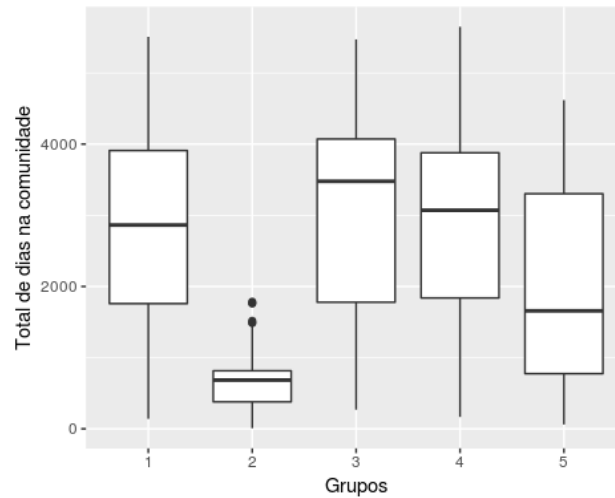
Esses fatos sugerem a hipótese de que os usuários desse grupo têm um objetivo de curto ou médio prazo, sejam eles: adquirir experiência, ganhar dinheiro rapidamente, executar alguma atividade relacionada à academia, entre outros; experimentar a plataforma; verificar se é viável investir tempo nela. Quando esses objetivos são alcançados, os usuários se desvinculam da plataforma. Essa hipótese explicaria a característica de começar a contribuir rapidamente e parar de participar da plataforma de modo precipitado.

### 5.3.6 Comparação entre os grupos

Esta seção compara os grupos com o intuito de evidenciar as características de cada um dos grupos e ressaltar suas diferenças. Para isso foram criados gráficos para cada uma das características utilizada na clusterização. Em alguns casos, outros atributos, que não os citados anteriormente, foram utilizados para auxiliar na compreensão do fenômeno referido.

A primeira comparação apresentada na Figura - 5.3 contrapõe a primeira característica utilizada na clusterização. Assim como retratado pelos atributos obtidos na Tabela - 5.1, os grupos elite, desistentes e tardios (grupos 1, 3 e 4, respectivamente), concentram os usuários que estão há mais tempo na plataforma. Por meio desse gráfico, fica evidente que os usuários com menos tempo de participação estão concentrados no grupo dos recém-chegados (grupo 2). O principal objetivo deste atributo é verificar a possibilidade de identificar diferenças entre usuários que estão há mais tempo na plataforma com os usuários que estão há menos tempo.

**Figura 5.3:** Distribuição dos usuários na característica dias na plataforma

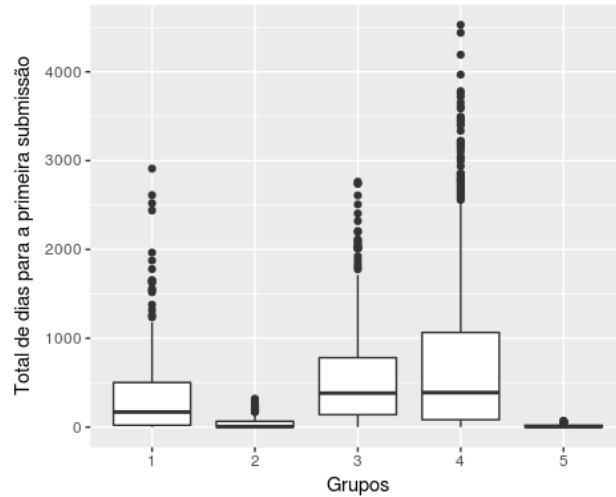


A segunda comparação é feita por meio da Figura - 5.4. Nela é comparado o total de dias para a primeira submissão na plataforma. É possível verificar que os grupos recém-chegados e usuários aventureiros (grupo 2 e 5 respectivamente) rapidamente começam a submeter suas contribuições, ao contrário dos outros grupos, que demoram muito mais para começar a contribuir. É oportuno destacar a quantidade de *outliers* nos grupos elite, desistentes e tardios (grupos 1, 3 e 4 respectivamente), mostrando que existem casos extremos em que os usuários demoram mais de 4500 dias para começar a submeter.

Uma observação pode ser feita quando os dados deste gráfico são relacionados às vitórias de cada grupo. Os usuários dos grupos recém-chegados e aventureiros (grupos 2 e 5 respectivamente) rapidamente começam a contribuir (submeter nos desafios), contudo possuem baixas taxas de vitórias, sendo que os usuários dos grupos elite e tardios (grupos 1 e 4 respectivamente) são os que mais demoram para começar a competir, porém eles possuem uma grande taxa de vitórias. Por fim, os usuários do grupo desistentes (grupo 3) possuem uma alta média de dias para começar a contribuir, contudo este grupo possui a pior média de vitórias entre os grupos.

A comparação feita na Figura - 5.5 mostra uma importante característica abordada anteriormente: a desistência dos usuários. Fica claro que o grupo dos usuários recém-chegados (grupo 2) possui a menor quantidade de dias entre a primeira e a última submissão. Este fato, potencialmente preocupante, foi analisado em uma outra perspectiva. Para ser justo na comparação em relação à quantidade de dias na comunidade que os usuários possuíam para competir, foi elaborada a Tabela - 5.4. Ela apresenta a quantidade de dias em competição dividido pela quantidade de dias na comunidade. Deste modo, tem-se a porcentagem de dias participados. Fica evidente que os usuários

**Figura 5.4:** Distribuição dos usuários na característica total de dias para a primeira submissão

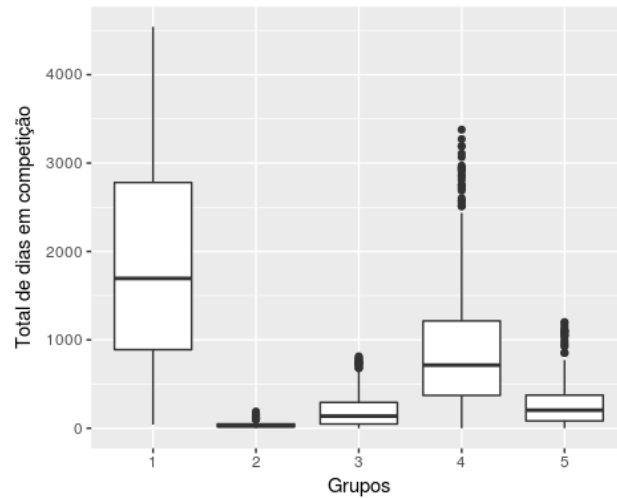


recém-chegados possuem a menor proporção de dias em competição, seguido de perto do grupo dos desistentes. Contudo, ainda é possível que os usuários dos novatos estivessem começando sua participação na plataforma. Para analisar esta hipótese foi observada a diferença em dias entre a data inicial do último desafio com submissão e a data de início da coleta. Para os usuários recém-chegados, a média da diferença de dias é 561,38 dias (86,64% do total de dias na plataforma), para os usuários desistentes é de 2301,01 dias (75,28% do total de dias na plataforma). Este resultado reforça a observação de que os usuários recém-chegados possuem a menor quantidade de dias na plataforma, pois foi visto que eles possuem tempo hábil para continuar a competir.

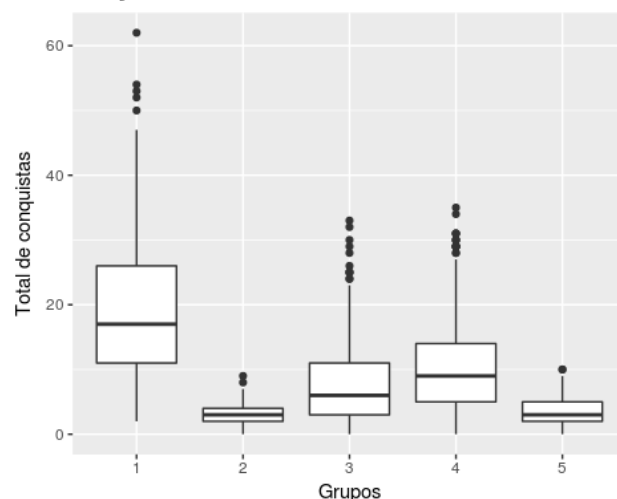
Continuando a análise da Figura - 5.5, uma característica ressaltada neste gráfico é a similaridade entre os grupos desistentes e aventureiros (grupos 3 e 5 respectivamente). Identificar os fatores que levam à ação de parar de contribuir é importante para a preservação do modelo. Por fim, os usuários que mais vencem, grupo da elite e dos tardios (grupos 1 e 4 respectivamente), possuem também mais tempo de competição.

**Tabela 5.4:** Proporção do tempo utilizado na plataforma para competir

Grupos	#Dias em competição/#Dias na plataforma
Grupo 1	0,6729%
Grupo 2	0,0618%
Grupo 3	0,0645%
Grupo 4	0,2991%
Grupo 5	0,1317%

**Figura 5.5:** Distribuição dos usuários na característica total de dias em competição

Uma característica pouco abordada anteriormente está representada na Figura - 5.6. Ela retrata o nível de interação dos usuários com a plataforma, justamente porque cada conquista indica que o usuário atingiu alguma marca como 100 postagens no fórum, primeira vitória, 50 primeiras submissões com sucesso entre outras. Como era visto na tabela dos centros da clusterização, o grupo que mais possui conquistas é o que mais vence o grupo dos usuários elite (grupo 1). Os grupos recém-chegados e aventureiros (grupos 2 e 5 respectivamente) possuem os níveis mais baixo de conquistas e possuem também o menor nível de vitórias.

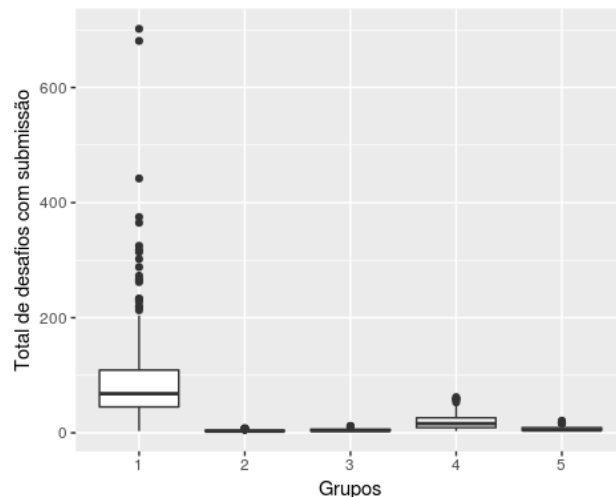
**Figura 5.6:** Distribuição dos usuários na característica total de conquistas

A Figura - 5.7 representa a quantidade de desafios com submissão dos grupos identificados. Uma relevante característica deste gráfico está presente na distribuição dos dados dos



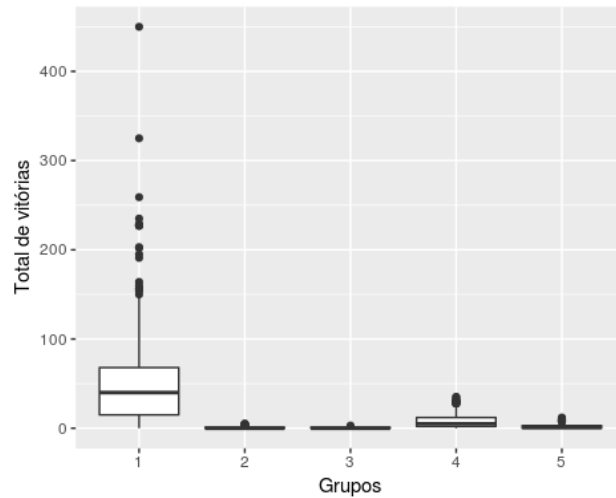
usuários da elite (grupo 1). Neste grupo é possível verificar que existem usuários *outliers* que possuem mais de 200 desafios com submissão. Olhando diretamente nos dados, é possível verificar que existem 23 usuários que ultrapassam a marca de 200 desafios com submissão. Esses 23 *outliers* correspondem à 9% da amostra e possuem 30% de todos os desafios com submissão do grupo (7160 desafios) e ainda possuem 32% de todas as vitórias do grupo (4551 vitórias). Em relação aos demais grupos é, possível verificar que o dos usuários tardios (grupo 4) possui usuários com mais desafios com submissão do que os recém-chegados, desistentes e aventureiros (grupos 2, 3 e 5 respectivamente)

**Figura 5.7:** Distribuição dos usuários na característica total de desafios com submissão

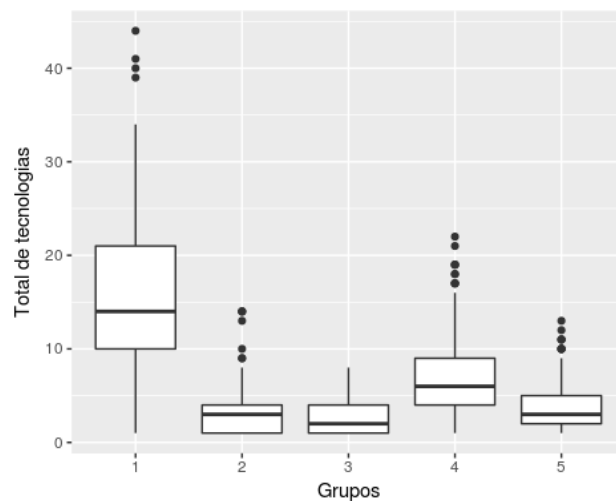


A distribuição das vitórias é uma importante característica utilizada nas análises e está retratada na Figura - 5.8. Assim como na figura anterior, existe certa dificuldade para analisar os grupos recém-chegados, desistentes, tardios e aventureiros (2, 3, 4 e 5 respectivamente), contudo, o que fica evidente que os usuários do grupo tardios (grupo 4) possuem mais vitórias que os usuários dos grupos recém-chegados, desistentes e aventureiros (2, 3 e 5 respectivamente), da mesma forma que o grupo dos usuários da elite (grupo 1) possui a maior distribuição das vitórias. Analisando os *outliers* do grupo 1, é possível verificar que 19 usuários (7% da população do grupo) possuem, no mínimo, 150 vitórias. Esses usuários são responsáveis por 28,9% de todas as vitórias do grupo.

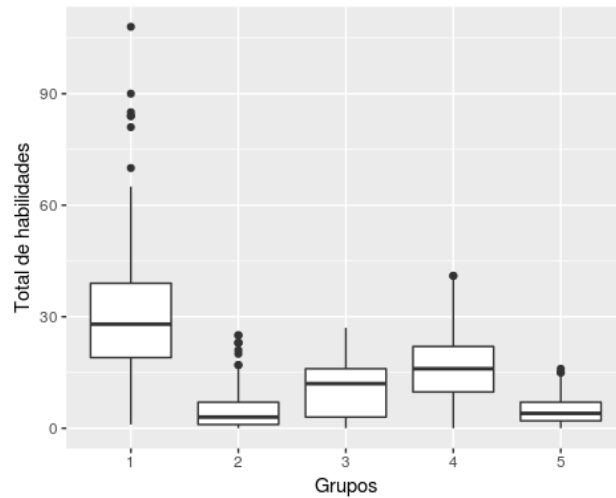
A característica total de tecnologias retratada na Figura - 5.9 foi utilizada também na análise relacionada à hiperespecialização. Ela indica diversidade de tecnologias presentes nos desafios de que os usuários participaram. Dados os resultados apresentados na fase anterior, era esperado que os grupos com maior diversidade de tecnologias fossem também os com maior número de vitórias (grupos elite e tardios). É oportuno destacar a presença

**Figura 5.8:** Distribuição dos usuários na característica total de vitórias

de *outliers* em quase todos os grupos, principalmente os usuários do grupo elite que possuem participação em mais de 35 tecnologias distintas.

**Figura 5.9:** Distribuição dos usuários na característica total de tecnologias

A distribuição do total de habilidades entre os grupos retratado pela Figura - 5.10 foi utilizada para confirmar as observações verificadas na figura anterior. Essa é uma informação que os usuários podem alterar conforme sua vontade. Apesar da quantidade de habilidades informadas serem, em geral, o dobro das tecnologias participadas nos desafios, sua distribuição entre os grupos é bem similar à distribuição das tecnologias. Um motivo para isso pode ser o fato de ter sido preservada a quantidade de habilidade original sem nenhum tipo de agrupamento por semelhança como feito no caso das tecnologias nos desafios.

**Figura 5.10:** Distribuição dos usuários na característica total de habilidades

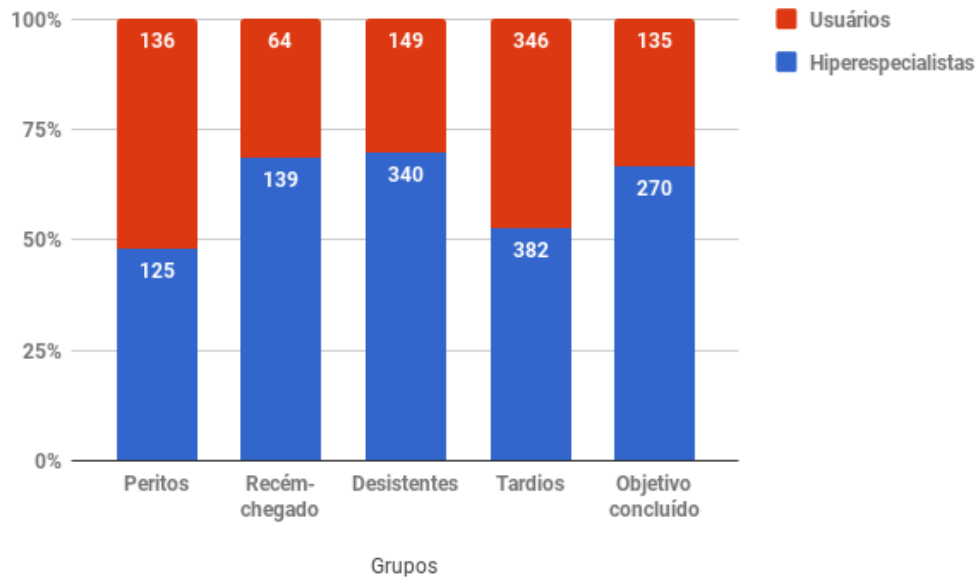
Por fim, a Figura - 5.11 representa a distribuição dos usuários classificados como hiperespecialistas nos grupos. A primeira observação é que existem menos hiperespecialistas nos grupos de usuários que mais possuem vitórias (grupos tardios e elite). Este resultado era esperado devido aos resultados da Fase 1 indicarem que os usuários hiperespecialistas possuem menos vitórias que os não especialistas. Além disso, outra importante observação é que, proporcionalmente, os grupos que mais possuem usuários com essa característica são os que possuem menor tempo de interação com a plataforma (grupos recém-chegados, desistentes e os aventureiros). Os usuários desses grupos possuem baixa taxa de dias em competição, baixo número de desafios com submissão, baixo número de vitórias e baixa quantidade de conquistas. Essas observações são comparáveis às observações obtidas na etapa 2 da análise da hiperespecialização (Fase 1), em que se observou que os usuários hiperespecialistas possuíam menos participações em desafios e vencem menos.

### 5.3.7 Implicações

Assim como verificado na etapa da análise da hiperespecialização, os resultados obtidos por essa etapa possuem implicações em algumas áreas.

**Empresas.** A existência de grupos de usuários elite (grupo 1) composto por desenvolvedores com muitas submissões, vitórias, habilidades e experiência (dias na plataforma) caracterizam um grupo de usuários qualificado para completar as tarefas propostas pelos clientes. Deste modo, este resultado se torna válido para as empresas, pois sabendo da existência de mão de obra qualificada as empresas podem confiar o desenvolvimento de seus projetos ao *crowdsourcing*.

**Figura 5.11:** Distribuição dos usuários na característica hiperespecialista



**Pesquisadores.** Alguns resultados encontrados neste trabalho podem ser explorados em outras pesquisas:

- Verificar quais fatores além da experiência (quantidade de dias na plataforma) e diversidade de conhecimento (quantidade de tecnologias e habilidades) são importantes para formarem usuários competitivos que participam e vencem muitos desafios.
- Verificar se algum fator motivacional como dinheiro, fama, oportunidade profissional ou passatempo tem mais influência sobre os usuários da elite;
- Identificar quais são os fatores que induzem os usuários a parar de competir, resultado encontrado na etapa 1 da análise da hiperespecialização e corroborado com a existência dos grupos desistente, recém-chegados e aventureiros;
- Verificar se é possível identificar usuários potencialmente aptos para se tornarem copilotos, usuários que auxiliam a empresa cliente ao longo do projeto;
- Verificar se entre os perfis de usuários existem perfis que pertencem a empresas, fazendo-os ter modos de participações diferentes.

**Plataforma.** Como identificado por meio dos grupos desistentes, recém-chegados e aventureiros, alguns usuários começam a participar e param de contribuir. Seria oportuno

criar estratégias para reter ou trazer de volta alguns usuários para potencialmente aumentar a quantidade de usuários aptos a competir, diversificando os conhecimentos e experiências. Enviar e-mails mostrando casos reais de sucesso para usuários que há a certo tempo não competem ou enviar tarefas disponíveis para os usuários conforme as habilidades informadas são ações que poderiam ser adotadas para tentar resgatar alguns usuários que não voltam à plataforma. Para usuários que acessam a plataforma, uma medida que poderia auxiliar na retenção seria separar as tarefas recomendadas para novos usuários e as indicadas para desenvolvedores experientes. É possível ainda, pensar em estratégias para que os usuários que mais se beneficiem da plataforma (elite, por exemplo), sigam motivados.

## 5.4 Limitações e ameaças a validade

Assim como na análise da hiperespecialização, o principal ponto que pode vir a ser discutido é a amostra. Como grande parte da população disponível possui, relativamente, pouca interação com a plataforma, ela foi reduzida para que os resultados pudessem representar usuários que efetivamente contribuem com a plataforma.

Especificamente em relação à análise dos dados, o uso dos centros das características como atributo definidor dos grupos pode ser questionado. No trabalho de Zanatta et al. (2017) foram utilizados os centróides para a separação dos usuários, assim como feito neste trabalho. Por meio da Tabela - 1.6, disponível no Apêndice A, é possível verificar que os valores das médias e dos centros são muito próximos. Desse modo, não haveria diferença significativa na representação dos grupos, caso este fosse o atributo utilizado para a definição dos mesmos. Além disso, todas as comparações feitas na Seção 5.3.6 confirmam as análises feitas. Por fim, quando possível, as observações tiveram o respaldo de atributos que não foram utilizados para a criação dos grupos.

A experiência com a utilização do algoritmo de clusterização denominado *Model-based Clustering* também é passível de discussão, porque nem sempre a ação de encontrar características comuns em grupos é trivial. Como frisado anteriormente, para que este algoritmo gere a clusterização, não é necessário nenhuma informação adicional, como a quantidade de *clusters* ou os *centroids* iniciais. Desse modo, somente as informações relacionadas aos usuários são necessárias para a realização do processo. Contudo, durante os testes realizados foi possível verificar que, com a adição ou remoção de apenas um dos atributos, os resultados eram, em geral, bem diferentes. Deste modo, foi decidido quais seriam os atributos utilizados e os resultados foram analisados. Além disso, é necessário cautela ao analisar os resultados dos grupos obtidos. Pois a diferença entre os atributos

dos grupos é sutil e nem sempre os identificados representam fielmente todos os tipos de usuários presentes na plataforma.

## 5.5 Conclusão

Por meio dos resultados apresentados anteriormente, a resposta para a questão de pesquisa dois (QP2) *Baseado nas ações dos usuários, é possível identificar grupos de perfis de usuários na plataforma Topcoder?* é sim. Os 2.086 usuários tiveram algumas de suas características contadas e foram separados em cinco grupos cada qual com sua característica.

Dentre os grupos encontrados tem-se o dos recém-chegados na plataforma. As características deste vão de encontro com o que a literatura retrata. Fica evidente a dificuldade que esses usuários possuem para se firmarem na plataforma, bem como a necessidade de ações para mantê-los contribuindo. Outro importante grupo encontrado, e que vai de encontro diretamente com os resultados da análise da hiperespecialização, é o grupo dos usuários da elite. Este é o grupo de usuários que mais vence, que mais submete e que possui a maior variedade de tecnologias.

Por fim, os resultados desta seção ajudam na compreensão de como contribuem os usuários mais assíduos da plataforma de *crowdsourcing* chamada Topcoder. Mostrando pontos que necessitam de atenção para garantir que o modelo continue evoluindo para sempre fornecer software de qualidade e uma opção de trabalho remunerado.

---

## Conclusão

---

Considerando que a literatura não enuncia todas as características acerca do perfil de contribuição dos participantes de *crowdsourcing*, em específico os que contribuem com a plataforma Topcoder, o objetivo deste trabalho é auxiliar na compreensão dos perfis de contribuição dos participantes da plataforma baseado em suas ações. Desse modo, foram conduzidos análises com o intuito de identificar e classificar os diferentes perfis de contribuição.

Uma hipótese atrelada a alguns resultados encontrados neste trabalho é a possível dificuldade encontrada por usuários, novos ou não, de interagir com a plataforma e encontrar as melhores opções para participar do *crowdsourcing*. Como citado anteriormente, alguns trabalhos haviam reportado tais dificuldades. O trabalho de Campo et al. (2018) fornece mais um indício de que essa preocupação é relevante. Neste trabalho, foram analisadas 20 plataformas de *crowdsourcing* inclusive a Topcoder. Os resultados, em geral, demonstram que todas as plataformas necessitam de melhorias, porém a Topcoder é uma das que obteve melhores resultados.

Como foi apresentado no Capítulo 4, foi possível identificar hiperespecialistas. Além disso, foi possível verificar que os hiperespecialistas, em geral, acabam competindo e vencendo menos. Uma resultado interessante (e preocupante) é que muitos usuários param de competir prematuramente. Além disso, não foi possível identificar hiperespecialistas para algumas tecnologias importantes e conhecidas como MySQL e PostgreSQL.

Na segunda fase, foram identificados cinco grupos de usuários que participam de desafios do tipo desenvolvimento: elite, recém-chegados, desistentes, tardios e usuários que atingiram seu objetivo. Por meio dessa análise, foi possível reforçar pontos como:

dificuldade em manter certos usuários participando; a existência de um grupo que vence grande parte dos desafios; a existência de novos usuários. Os resultados desta fase corroboram os resultados obtidos na fase anterior e vão de encontro com a literatura existente.

## 6.1 Lições aprendidas

Uma das maiores dificuldades encontradas ao longo deste trabalho está relacionada à integridade e consistência dos dados coletados, justamente para que os resultados encontrados pudessem representar fielmente a realidade existente no ecossistema. Deste modo, uma alteração possível seria utilizar um banco de dados não relacional como, por exemplo, o MongoDB, ao invés de usar um banco de dados relacional como foi utilizado ao longo deste trabalho. Com essa alteração seria possível armazenar todas as informações presentes nas respostas de cada requisição, evitando assim, a falta de informações que, em um segundo momento, seriam necessárias.

Além da alteração do tipo do banco de dados, outra alteração interessante seria encontrar uma forma mais eficiente de agrupar os usuários baseados em seus diferentes perfis de contribuição na fase 2 deste trabalho. Para tal resultado, foi utilizado um algoritmo que não necessita de configuração inicial, como quantidade de agrupamentos esperados ou centros iniciais. Por meio dos testes efetuados, foi verificado que, tal abordagem pode não ser a mais indicada para o contexto, sendo possível que outra abordagem possa apresentar resultados mais fidedignos para o ecossistema.

## 6.2 Trabalhos futuros

Será planejado um próximo trabalho mais qualitativo que tem por objetivo explorar mais detalhes sobre o perfil dos hiperespecialistas para entender as estratégias, os benefícios e as desvantagens de ser um especialista.

Apesar de algumas características a respeito do perfil de contribuição dos usuários terem sido esclarecidas neste trabalho existem contribuições importantes que precisam ser feitas. Uma delas é buscar motivos concretos para a alta taxa de desistência de usuários na plataforma. Essa característica foi encontrada na fase relacionada à hiperespecialização e à caracterização dos perfis. Compreendendo melhor o êxodo dos participantes de *crowdsourcing* é possível sugerir ações para mantê-los contribuindo, objetivando a criação de software de qualidade e absorvendo as vantagens do modelo. Além disso, seria interessante



aprofundar nas características dos grupos de usuários que mais vencem (usuários da elite) buscando identificar os motivos que os levaram ao topo da plataforma.

Também é de interesse mostrar que o *crowdsourcing* pode ser utilizado em conjunto com outras técnicas/áreas da engenharia de software. Um exemplo disto seria a possibilidade de utilizar a comunidade para desenvolver os módulos de linha de produto de software, uma vez que, em uma linha de produto de software, as partes integrantes do sistema são feitas de modo que seja possível remover uma parte e adicionar outra com o intuito de gerar um novo produto.

Por fim, seria de grande valia traçar características comuns e diferenças entre as comunidades de desenvolvedores que atuam em plataformas de *crowdsourcing* e de software livre; tentar verificar se existe uma migração entre elas, ou se os grupos são compostos pelos mesmos usuários, ou são compostos por desenvolvedores diferentes, por exemplo.

# REFERÊNCIAS

---

- ABHINAV, K.; DUBEY, A.; VIRDI, G.; KASS, A. Analyzing on-boarding time in context of crowdsourcing. In: *Proceedings of the 2nd International Workshop on Software Analytics*, ACM, 2016, p. 29–35.
- ALVERTIS, I.; PAPASPYROS, D.; KOUSSOURIS, S.; MOUZAKITIS, S.; ASKOUNIS, D. Using crowdsourced and anonymized personas in the requirements elicitation and software development phases of software engineering. In: *Availability, Reliability and Security (ARES), 2016 11th International Conference on*, IEEE, 2016, p. 851–856.
- AMBREEN, T.; IKRAM, N. A state-of-the-art of empirical literature of crowdsourcing in computing. In: *Global Software Engineering (ICGSE), 2016 IEEE 11th International Conference on*, IEEE, 2016, p. 189–190.
- BREAUX, T. D.; SCHAUB, F. Scaling requirements extraction to the crowd: Experiments with privacy policies. In: *2014 IEEE 22nd International Requirements Engineering Conference (RE)*, 2014, p. 163–172.
- CAMPO, S.; KHAN, V.-J.; PAPANGELIS, K.; MARKOPOULOS, P. Community heuristics for user interface evaluation of crowdsourcing platforms. *Future Generation Computer Systems*, 2018.
- COOPER, S.; KHATIB, F.; TREUILLE, A.; BARBERO, J.; LEE, J.; BEENEN, M.; LEAVER-FAY, A.; BAKER, D.; POPOVIĆ, Z.; ET AL. Predicting protein structures with a multiplayer online game. *Nature*, v. 466, n. 7307, p. 756–760, 2010.
- DUBEY, A.; ABHINAV, K.; TANEJA, S.; VIRDI, G.; DWARAKANATH, A.; KASS, A.; KURIAKOSE, M. S. Dynamics of software development crowdsourcing. In: *Global Software Engineering (ICGSE), 2016 IEEE 11th International Conference on*, IEEE, 2016, p. 49–58.

DWARAKANATH, A.; SHRIKANTH, N.; ABHINAV, K.; KASS, A. Trustworthiness in enterprise crowdsourcing: a taxonomy & evidence from data. In: *Proceedings of the 38th International Conference on Software Engineering Companion*, ACM, 2016, p. 41–50.

FRALEY, C.; RAFTERY, A. E. Model-based clustering, discriminant analysis, and density estimation. *Journal of the American statistical Association*, v. 97, n. 458, p. 611–631, 2002.

FRALEY, C.; RAFTERY, A. E.; MURPHY, T. B.; SCRUCCA, L. mclust version 4 for r: Normal mixture modeling for model-based clustering, classification, and density estimation. *University of Washington: Seattle*, 2012.

Disponível em <https://www.stat.washington.edu/sites/default/files/files/reports/2012/tr597.pdf>

FURTADO, A.; ANDRADE, N.; OLIVEIRA, N.; BRASILEIRO, F. Contributor profiles, their dynamics, and their importance in five q&a sites. In: *Proceedings of the 2013 conference on Computer supported cooperative work*, ACM, 2013, p. 1237–1252.

GADIRAJU, U. Make hay while the crowd shines: Towards efficient crowdsourcing on the web. In: *Proceedings of the 24th International Conference on World Wide Web*, ACM, 2015, p. 493–497.

GADIRAJU, U.; SIEHNDEL, P.; FETAHU, B.; KAWASE, R. Breaking bad: Understanding behavior of crowd workers in categorization microtasks. In: *Proceedings of the 26th ACM Conference on Hypertext & Social Media*, ACM, 2015, p. 33–38.

GRAY, M. L.; SURI, S.; ALI, S. S.; KULKARNI, D. The crowd is a collaborative network. In: *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*, ACM, 2016, p. 134–147.

HASTEER, N.; NAZIR, N.; BANSAL, A.; MURTHY, B. Crowdsourcing software development: Many benefits many concerns. *Procedia Computer Science*, v. 78, p. 48–54, 2016.

HINKLE, D. E.; WIERSMA, W.; JURIS, S. G. *Applied statistics for the behavioral sciences 5th edition*. 2002.

HO, C.-J.; VAUGHAN, J. W. Online task assignment in crowdsourcing markets. In: *AAAI*, 2012, p. 45–51.

- HOSSAIN, M. Users' motivation to participate in online crowdsourcing platforms. In: *Innovation Management and Technology Research (ICIMTR), 2012 International Conference on*, IEEE, 2012, p. 310–315.
- HOSSEINI, M.; PHALP, K.; TAYLOR, J.; ALI, R. The four pillars of crowdsourcing: A reference model. In: *RCIS Conference 2014*, IEEE, 2014, p. 1–12.
- HOWE, J. Crowdsourcing: A definition. *Wired Magazine*, v. 14, n. 06, p. 1–4, 2006.
- KARIM, M. R.; MESSINGER, D.; YANG, Y.; RUHE, G. Decision support for increasing the efficiency of crowdsourced software development. *arXiv preprint arXiv:1610.04142*, 2016.
- KAUFMANN, N.; SCHULZE, T.; VEIT, D. More than fun and money. worker motivation in crowdsourcing-a study on mechanical turk. In: *AMCIS*, 2011, p. 1–11.
- KITTUR, A.; CHI, E.; PENDLETON, B. A.; SUH, B.; MYTKOWICZ, T. Power of the few vs. wisdom of the crowd: Wikipedia and the rise of the bourgeoisie. *World wide web*, v. 1, n. 2, p. 19, 2007.
- LASECKI, W. S.; KIM, J.; RAFTER, N.; SEN, O.; BIGHAM, J. P.; BERNSTEIN, M. S. Apparition: Crowdsourced user interfaces that come to life as you sketch them. In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, New York, NY, USA: ACM, 2015, p. 1925–1934 (*CHI '15*, ).  
Disponível em <http://doi.acm.org/10.1145/2702123.2702565>
- LATOZA, T. D.; VAN DER HOEK, A. Crowdsourcing in software engineering: Models, motivations, and challenges. *IEEE Software*, v. 33, n. 1, p. 74–80, 2016.
- LIU, D.; BIAS, R. G.; LEASE, M.; KUIPERS, R. Crowdsourcing for usability testing. *Proceedings of the Association for Information Science and Technology*, v. 49, n. 1, p. 1–10, 2012.
- LYDON, M. Topcoder overview. [https://www.nasa.gov/pdf/651447main\\_TopCoder\\_Mike\\_D1\\_830am.pdf](https://www.nasa.gov/pdf/651447main_TopCoder_Mike_D1_830am.pdf), 2012.
- MACHADO, L.; ZANATTA, A.; MARCZAK, S.; PRIKLADNICKI, R. The good, the bad and the ugly: an onboard journey in software crowdsourcing competitive model. In: *Proceedings of the 4th International Workshop on CrowdSourcing in Software Engineering*, IEEE Press, 2017, p. 2–8.

- MALONE, T. W.; LAUBACHER, R. J.; JOHNS, T. The age of hyperspecialization. *Harvard Business Review*, v. 89, n. 7-8, p. 56–+, 2011.
- MAO, K.; CAPRA, L.; HARMAN, M.; JIA, Y. A survey of the use of crowdsourcing in software engineering. *RN*, v. 15, n. 01, 2015a.
- MAO, K.; WANG, Q.; JIA, Y.; HARMAN, M. Prem: Prestige network enhanced developer-task matching for crowdsourced software development. 2016.
- MAO, K.; YANG, Y.; WANG, Q.; JIA, Y.; HARMAN, M. Developer recommendation for crowdsourced software development tasks. In: *Service-Oriented System Engineering (SOSE), 2015 IEEE Symposium on*, IEEE, 2015b, p. 347–356.
- MESSENGER, J.; VARGAS LLAVE, O.; GSCHWIND, L.; BOEHMER, S.; VERMEYLEN, G.; WILKENS, M. Working anytime, anywhere: The effects on the world of work, p. 80. 2017.
- MOK, R. K.; CHANG, R. K.; LI, W. Detecting low-quality workers in qoe crowdtesting: A worker behavior based approach. *IEEE Transactions on Multimedia*, 2016.
- MOK, R. K.; LI, W.; CHANG, R. K. Detecting low-quality crowdtesting workers. In: *2015 IEEE 23rd International Symposium on Quality of Service (IWQoS)*, IEEE, 2015, p. 201–206.
- PINTO, L. F. S.; DOS SANTOS, C. D. Motivations of crowdsourcing contributors. *Innovation & Management Review*, 2018.
- QIU, C.; SQUICCIARINI, A. C.; CARMINATI, B.; CAVERLEE, J.; KHARE, D. R. Crowdselect: Increasing accuracy of crowdsourcing tasks through behavior prediction and user selection. In: *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, ACM, 2016, p. 539–548.
- ROGSTADIUS, J.; KOSTAKOS, V.; KITTUR, A.; SMUS, B.; LAREDO, J.; VUKOVIC, M. An assessment of intrinsic and extrinsic motivation on task performance in crowdsourcing markets. *ICWSM*, v. 11, p. 17–21, 2011.
- SAREMI, R. L.; YANG, Y. Dynamic simulation of software workers and task completion. In: *Proceedings of the Second International Workshop on CrowdSourcing in Software Engineering*, IEEE Press, 2015, p. 17–23.

SAREMI, R. L.; YANG, Y.; RUHE, G.; MESSINGER, D. Leveraging crowdsourcing for team elasticity: an empirical evaluation at topcoder. In: *2017 IEEE/ACM 39th International Conference on Software Engineering: Software Engineering in Practice Track (ICSE-SEIP)*, 2017, p. 103–112.

DE SOUZA, L. B. L.; CAMPOS, E. C.; MAIA, M. D. A. Ranking crowd knowledge to assist software development. In: *Proceedings of the 22Nd International Conference on Program Comprehension, ICPC 2014*, New York, NY, USA: ACM, 2014, p. 72–82 (*ICPC 2014*, ).

Disponível em <http://doi.acm.org/10.1145/2597008.2597146>

STOL, K. J.; LATOZA, T. D.; BIRD, C. Crowdsourcing for software engineering. *IEEE Software*, v. 34, n. 2, p. 30–36, 2017.

SUROWIECKI, J. The wisdom of crowds: why the many are smarter than the few. london. *Abacus: New Edition*, 2005.

THELWALL, M. Social networks, gender, and friending: An analysis of myspace member profiles. *J. Am. Soc. Inf. Sci. Technol.*, v. 59, n. 8, p. 1321–1330, 2008.

Disponível em <http://dx.doi.org/10.1002/asi.v59:8>

TONG, Y.; CHEN, L.; ZHOU, Z.; JAGADISH, H. V.; SHOU, L.; LV, W. Slade: A smart large-scale task decomposer in crowdsourcing. *IEEE Transactions on Knowledge and Data Engineering*, p. 1–1, 2018.

TUNIO, M. Z.; LUO, H.; CONG, W.; FANG, Z.; GILAL, A. R.; ABRO, A.; WENHUA, S. Impact of personality on task selection in crowdsourcing software development: A sorting approach. *IEEE Access*, v. 5, p. 18287–18294, 2017.

XIE, H.; LUI, J. C.; JIANG, J. W.; CHEN, W. Incentive mechanism and protocol design for crowdsourcing systems. In: *Communication, Control, and Computing (Allerton), 2014 52nd Annual Allerton Conference on*, IEEE, 2014, p. 140–147.

XU, A.; FENG, X.; TIAN, Y. Revealing, characterizing, and detecting crowdsourcing spammers: A case study in community q&a. In: *2015 IEEE Conference on Computer Communications (INFOCOM)*, IEEE, 2015, p. 2533–2541.

YANG, Y.; SAREMI, R. Award vs. worker behaviors in competitive crowdsourcing tasks. In: *2015 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*, IEEE, 2015, p. 1–10.

YE, B.; WANG, Y. Crowdrec: Trust-aware worker recommendation in crowdsourcing environments. In: *Web Services (ICWS), 2016 IEEE International Conference on*, IEEE, 2016, p. 1–8.

YUEN, M.-C.; KING, I.; LEUNG, K.-S. Task recommendation in crowdsourcing systems. In: *Proceedings of the first international workshop on crowdsourcing and data mining*, ACM, 2012, p. 22–26.

ZANATTA, A. L.; STEINMACHER, I.; MACHADO, L. S.; DE SOUZA, C. R.; PRIKLADNICKI, R. Barriers faced by newcomers to software-crowdsourcing projects. *IEEE Software*, v. 34, n. 2, p. 37–43, 2017.

ZHANG, D.; PRIOR, K.; LEVENE, M. How long do wikipedia editors keep active? In: *Proceedings of the Eighth Annual International Symposium on Wikis and Open Collaboration*, ACM, 2012, p. 4.

ZOU, G.; GIL, A.; THARAYIL, M. An agent-based model for crowdsourcing systems. In: *Proceedings of the 2014 Winter Simulation Conference*, IEEE Press, 2014, p. 407–418.

# Apêndice A

---

## A.1 Dados dos grupos

O intuito deste apêndice é complementar os resultados obtidos na fase da análise dos perfis de usuário, reportando outras métricas sobre cada um dos grupos. As informações apresentadas neste apêndice são: mínimo, máximo, primeiro quartil, terceiro quartil e centros de agrupamento. A Tabela - 1.2 reporta os menores valores encontrados para cada um dos atributos para todos os grupos. A Tabela - 1.3 reporta o maior valor observado para todos os grupos. A Tabela - 1.4 e a Tabela - 1.5 reportam o limiar do primeiro e do terceiro quartil para os grupos. Por fim, a Tabela - 1.6 mostra quais são as médias para cada um os atributos.

Assim como feito no Capítulo 5, o nome completo dos atributos foram substituídos por abreviações. Por exemplo a primeira característica apresentada na Seção 5.2 é o “Total de dias na comunidade” que foi substituído por “C1”, a segunda característica foi “Total de dias para a primeira submissão” que foi substituída por “C2” e assim para todos os outros atributos. Todas as abreviações estão disponíveis na Tabela - 1.1, reproduzida novamente para auxiliar na leitura dos dados.

**Tabela 1.1:** Relação das abreviaturas com os nomes completos das características

Abreviação	Nome completo
C1	Total de dias na comunidade
C2	Total de dias para a primeira submissão
C3	Total de dias em competição
C4	Total de conquistas
C5	Total de desafios com submissão
C6	Total de vitórias
C7	Total de tecnologias
C8	Total de habilidades



**Tabela 1.2:** Valor mínimo para cada atributo dos grupos encontrados

Grupos	C1	C2	C3	C4	C5	C6	C7	C8
Grupo 1	141,00	0,00	41,00	2,00	3,00	0,00	1,00	1,00
Grupo 2	8,00	0,00	0,00	0,00	3,00	0,00	1,00	0,00
Grupo 3	269,00	0,00	0,00	0,00	3,00	0,00	1,00	0,00
Grupo 4	166,00	0,00	0,00	0,00	3,00	0,00	1,00	0,00
Grupo 5	61,00	0,00	0,00	0,00	3,00	0,00	1,00	0,00

**Tabela 1.3:** Valor máximo para cada atributo dos grupos encontrados

Grupos	C1	C2	C3	C4	C5	C6	C7	C8
Grupo 1	5508,00	2910,00	4542,00	62,00	702,00	450,00	44,00	108,00
Grupo 2	1786,00	319,00	189,00	9,00	8,00	5,00	14,00	25,00
Grupo 3	5472,00	2763,00	813,00	33,00	12,00	3,00	8,00	27,00
Grupo 4	5649,00	4529,00	3378,00	35,00	62,00	35,00	22,00	41,00
Grupo 5	4619,00	70,00	1201,00	10,00	21,00	12,00	13,00	16,00

**Tabela 1.4:** Primeiro quartil para cada atributo dos grupos encontrados

Grupos	C1	C2	C3	C4	C5	C6	C7	C8
Grupo 1	1759	21	889	11	45	15	10	19
Grupo 2	379	0	15,5	2	3	0	1	1
Grupo 3	1778	140	48	3	3	0	1	3
Grupo 4	1838,75	82,75	371,75	5	9	2	4	9,75
Grupo 5	775	0	83	2	4	0	2	2

**Tabela 1.5:** Terceiro quartil para cada atributo dos grupos encontrados

Grupos	C1	C2	C3	C4	C5	C6	C7	C8
Grupo 1	3910	502	2780	26	109	68	21	39
Grupo 2	815	65,5	49	4	4	1	4	7
Grupo 3	4072	781	292	11	6	1	4	16
Grupo 4	3878,5	1065	1213,75	14	26	12	9	22
Grupo 5	3303	20	374	5	9	3	5	7

**Tabela 1.6:** Médias dos grupos encontrados

Grupos	C1	C2	C3	C4	C5	C6	C7	C8
Grupo 1	2.804,85	357,86	1.889,77	19,23	90,75	54,82	15,75	30,41
Grupo 2	647,93	45,78	40,05	2,83	3,84	0,76	3,31	5,02
Grupo 3	3.056,22	558,73	195,71	7,94	4,81	0,42	2,69	10,31
Grupo 4	2.888,68	752,26	870,88	10,24	18,57	7,56	7,06	15,82
Grupo 5	2.081,46	12,21	275,16	3,57	6,84	1,89	3,80	4,83

# Apêndice B

---

## B.1 Banco de dados

A Figura - 2.1 representa todas as tabelas e os relacionamentos do banco de dados criado para dar suporte à este trabalho. Para melhor visualização da figura, a arquivo original está disponível neste endereço<sup>1</sup>.

---

<sup>1</sup>[https://bitbucket.org/anderson-neira/dissertacao/src/master/banco/banco\\_de\\_dados.png](https://bitbucket.org/anderson-neira/dissertacao/src/master/banco/banco_de_dados.png)

Figura 2.1: Digrama de entidade e relacionamento do banco de dados

