Decoding the Diversity of the German Software Developer Community: Insights from an Exploratory Cluster Analysis

Katharina Dworatzyk(✉), Vincent Dekorsy, and Sabine Theis

German Aerospace Center (DLR), Institute for Software Technology, Cologne, Germany
{katharina.dworatzyk,vincent.dekorsy,sabine.theis}@dlr.de

Abstract. To better understand the nature of the software developer community in Germany, an open data set of 2,837 responses collected through the Stack Overflow Developer Survey was analyzed. Cluster analysis identified four sub-groups that were characterized by significant differences in sociodemographic variables, programming experience and software development specialization: the web developer (25%), the junior developer (26%), the senior developer (39%), and the research developer (10%). Cluster characteristics were associated with differences in experienced community membership, frequency of Stack Overflow visits and participation in Q&A. Senior developers with the highest level of coding experience reported the overall strongest community engagement, while research developers visited Stack Overflow least often. A positive moderate relationship between community membership and community visits and community participation provided additional evidence for the importance of the sense of (virtual) community as a psychological driving factor of community engagement. Together, these findings can contribute to developing more targeted training and community-building measures.

Keywords: software developer · virtual community · personas · cluster analysis

1 Introduction

Software engineering is a dynamic, evolving field with continual advancements in technologies, methodologies, and best practices. At the core of this evolution are the software developer communities that provide a platform for sharing knowledge and learning about technological advancements. These communities refer to a collective of individuals and organizations involved in the creation, development, use, and maintenance of software. Various forms of communities, including open-source software communities [55], local meetups [22], and members of knowledge platforms or portals [16,23], provide opportunities for knowledge exchange, which is crucial for software engineers to stay updated and competitive. In addition, they are crucial for software engineers aiming to stay abreast of

H. Mori and Y. Asahi (Eds.): HCII 2024, LNCS 14691, pp. 275–295, 2024.
https://doi.org/10.1007/978-3-031-60125-5_19

technological advancements, find mentorship, collaborate, and build supportive networks [6,27].

Recent research underscores the significance of diversity within these communities. Studies indicate that understanding the composition of a community and promoting engagement and inclusion is essential to creating supportive, sustainable, and vibrant communities [25,38,45]. Semi-structured interviews, for example, highlighted that inclusion and diversity in terms of gender, ethnicity, disabilities, and age foster creativity [?]. Also, workforce diversity positively impacts on organizational competitiveness by increasing employee skills and talent, with a moderately positive relationship between age diversity and employee performance [52]. Furthermore, literature reviews showed that women's participation in open-source software (OSS) and the demographics of software development teams have a positive impact on community engagement [50], while especially increasing age diversity has a positive effect on company productivity if the company engages in creative rather than routine tasks [52]. Studies like these demonstrate how diversity regarding demographic characteristics and individual skills within a community significantly influences its quality.

In the realm of tool development for software engineers, the emphasis on understanding the community's diverse needs is also of paramount importance. The variety of tools used by developers reflects the diversity of their tasks, skills, and preferences [13]. By tailoring tools to meet these varied requirements, both functional and non-functional, the productivity and efficiency of developers can be significantly improved [26,47]. By understanding the community and its needs, working tools can be better adapted to requirements [20]. This can increase developer productivity and efficiency [9,11]. In general, creating efficient and effective software tools and workplaces benefits from studying users' needs and requirements [10]. One popular method to model different user types is constructing personas [30]. Personas are fictional characters representing common types of users and typically consist of a name, usually a photo, and demographic characteristics, skills, and abilities [5]. Designers, developers, and human factors experts apply them to get a clearer understanding of the target group they are designing products and workplaces for [8,12,28,34]. While in practice, personas are often developed qualitatively or purely on the basis of subjective perception, there are already the first successful approaches to taking quantitative data into account [40,41]

Given the critical role of diversity in fostering robust software engineering communities and the necessity of grasping user groups' diverse needs for effective tool development, a detailed examination of the current state of the software developer community, in particular in Germany, is imperative. This study aims to decode the diversity within this community through exploratory cluster analysis, seeking insights into how demographic characteristics, individual skills, and diversity factors influence community culture, engagement, and the potential for innovation.

2 Related Work

The following section provides an overview of research related to software engineering communities in Germany: the community characteristics, the role of diversity in software engineering in general, and its specific impact on this discipline. The aim is to get a clearer picture of how diversity is shaped within the German software engineering community and what impact this has.

2.1 Software Engineering Communities in Germany

In Germany, the importance of software engineering is increasing due to ongoing digitization. In 2022, the IT segment in Germany was the most profitable, with a turnover of around 46 billion euros, followed by the hardware and software sectors. Currently, there are over 90,000 companies in the IT industry with around 1.1 million employees. Nearly 35.5 billion euros were generated with software products in Germany in 2022[1]. The largest software companies include SAP, with an annual turnover of 24.71 billion euros in 2018, and DATEV, with 978 million euros in 2017, and Software AG, with an annual turnover of 879 million euros in 2017 [3]. But also many small and highly specialized software companies are based in Germany, for example, in the environment of groups in the automotive industry[2].

Software engineering takes place not only in industry but also in the research sector. While software engineering in research was largely overlooked for a long time, currently there are increasing efforts, for example in Germany, the Netherlands and the UK, to build a lively and growing research software engineering community. In contrast to previous work on software engineering, there is far more work in research software engineering that focuses on German samples. Analysis of data based on workshop participation [43,44] showed, for example, that the software development community of the DLR[3] consists of a growing core group regularly joined by new participants. Both software engineering and research software engineering communities consist of individual groups emerging or consolidating each other primarily through international platforms and national as well as international associations.

One of the largest and best-known platforms where software engineers, as well as research software engineers, can ask and answer questions to solve programming problems is Stack Overflow[4]. The asked questions and answers can be viewed by anyone registered, while all its content is licensed under Creative Commmons. Additionally, registered users of the platform are able to rate given answers so that they can be ranked and displayed accordingly. As an international platform, it has over 14 million registered users. In one study, data visualization was applied to gain insights into the international community of

[1] https://de.statista.com/themen/1373/it-branche-deutschland/#topicOverview.
[2] https://www.wlw.de/de/inside-business/aktuelles/deutsche-software.
[3] www.dlr.de.
[4] www.https://stackoverflow.com/.

Stack overflow and their development over time and to identify sub-communities, their sizes, and tools used by each community. The authors identified the web development community as the biggest and most persistent one over time. In general, community members exhibit low fluctuation rates between identified sub-communities [51]. In addition, Srba and Bielikova [48] concentrated on identifying groups based on individual skills and behavior. Publicly available data on user activity was used to study the quality of contributions over time, considering how knowledgeable users were. Results indicated that when more low-quality content was posted-particularly by users who frequently ask questions without looking for answers first (known as "help vampires"), beginners who ask many basic questions ("noobs"), and users who answer many questions to gain reputation ("reputation collectors")-this lead to higher turnover rates in the community. Similarly but more in-depth, Anderson and colleagues [4] investigated that a strong relationship between the key characteristics of a question and its answers have a long-term value for a community.

Stack Overflow studies that focus more on individual factors primarily look at behavior in relation to human factors such as personality and gender. Ahmed and Srivastava [2], for example, showed that community members preferred to answer to a person from the same locality or gender. Previous work on the Stack Overflow community has very often considered the use of technical tools, the identification of sub-groups in the community, and the behavior of these subgroups, as well as the quality of content and its impact on the community as a whole. Although initial studies show the influence of demographic characteristics such as origin or gender, there is a lack of detailed consideration of other diversity factors, particularly for the German developer community.

Like Stack Overflow, GitHub[5] is a popular community platform where especially open-source developers can exchange ideas and share their projects. This platform is often used to investigate the underlying community due to its extensive data set, but when interpreting the results, it must be borne in mind that this is a very technical user group [24]. On the other hand, GitHub data was used to examine collaboration in communities and teams. For example, to look at their development over time and to examine the influence of certain actors within a community on a development project in more detail [18]. The research utilizes GitHub data to explore how stakeholder involvement contributes to developing innovative solutions for environmental issues, emphasizing the critical role of community structures that facilitate effective communication and collaboration [31]. Similarly, Moutidis and Williams [32] conducted a study on Stack Overflow, identifying significant communities, examining the technologies these communities are interested in, and how they connect and evolve over time. These studies highlight the utility of data from large developer platforms like GitHub and Stack Overflow, offering insights into global software development practices. However, the focus on code(ing) of the GitHub platform naturally leads to corresponding data, analyses of which are less likely to provide knowledge about human characteristics, behavior, or diversity factors within the community. Consequently,

[5] www.github.com.

research specifically targeting the German developer community is limited, with only a few small-scale studies focusing on research software engineers. This gap points to a potential area for further research to better understand the engagement and contributions of the German developer community on these platforms.

Software engineering and developer communities in Germany are not only shaped by companies, research institutions, and international platforms but also by non-profit professional organizations. Notably, in Germany, organizations such as the Society for Informatics (Gesellschaft für Informatik, GI)[6], the Bundesverband IT-Mittelstand e.V. (BITMi)[7], which represents over 2,500 IT companies, play a significant role. The GI, in particular, has implemented robust initiatives to foster equal opportunities and gender equality both within its ranks and in the broader community GI [15]. However, up to now, their and other investigations of diversity in the developer community in Germany have concentrated on the proportion of women and girls in computer science education [42, 46]. The actions of the German research software engineering community signify a pivotal shift, bridging the focus on diversity initiatives by professional organizations such as the GI and BITMi with broader academic and research practices. This advocacy has culminated in the German Research Foundation (DFG) revising its guidelines to acknowledge software development on par with traditional academic outputs. The development of the German software developer community had already been strengthened years before by the founding of the "Distributed Competence Centers for Software Engineering (ViSEK)" [19]. This project concentrated on intra- and inter-organizational learning in the software industry by building communities among software engineers through an internet portal and a regional network of SMEs.

Analyzing interactions and contributions on platforms like GitHub or Stack Overflow can offer insights into the collaborative dynamics and diversity of the software engineering community [48, 51]. Leveraging open data sets related to software development projects allows for the examination of team composition, contribution patterns, and project outcomes in relation to diversity factors [48, 51]. Employing data visualization techniques can help in identifying patterns and trends related to diversity within software engineering, facilitating a deeper understanding of its impact [51]. Conducting case studies on specific software engineering teams or projects can provide detailed insights into how diversity influences team dynamics and project success [48]. These qualitative and quantitative approaches allow for a nuanced exploration of diversity within the software engineering community, enabling the identification of unique diversity clusters and the analysis of their influence on software development practices. By applying these methodological approaches, researchers can decode the complexities of diversity within the German software developer community, contributing to a richer understanding of its impacts and benefits in the field of software engineering.

---

6 www.gi.de.

7 https://www.bitmi.de/.

In summary, the German software engineering community is shaped by the national software engineering industry, research institutes, and professional and industry associations. Efforts to build German communities have been and are currently being made by public research funding as well as research software engineering initiatives or associations. Investigations of the German software engineering community are sparse and lack generalizability. The present work aims, therefore, to explore the diversity within the German software developer community through exploratory cluster analysis. By examining demographic characteristics, individual skills, and diversity factors, we seek to provide insights into how these elements influence community culture, engagement, and the potential for innovation. Our analysis contributes to a deeper understanding of the composition of the German software developer community, offering implications for fostering more inclusive, engaging, and productive communities in the software engineering field.

3 Materials and Methods

3.1 Stack Overflow Developer Survey

To improve our understanding of the different backgrounds and development tools of German developers, an open-access data set, resulting from the annual Stack Overflow Developer Survey[8], was analyzed. The survey was conducted from mid-May to the beginning of June 2022 and mainly advertised through Stack Overflow channels.

The complete data set contained responses to questions on (1) Basic Information about participants, e.g. "Which of the following options best describes you today? Here, by 'developer' we mean 'someone who writes code'." with the response options "I am a developer by profession", "I am not primarily a developer, but I write code sometimes as part of my work", "I used to be a developer by profession, but no longer am", "I am learning to code", "I code primarily as a hobby", "None of these"; (2) Education, Work, and Career, e.g. "Which of the following describes your current job?" with response options such as "Academic researcher", "Data or business analyst", "Developer, full-stack", "DevOps specialist" or "Project manager"; (3) Technology and Tech Culture, e.g. "Which programming, scripting, and markup languages have you done extensive development work in over the past year?" with multiple response options; (4) Stack Overflow Usage and Community, e.g. "Do you consider yourself a member of the Stack Overflow community?" with response options ranging from "No, not at all" to "Yes, definitely"; (5) Demographic Information, e.g. "What is your age?"; (6) Professional Development, e.g. "Are you an independent contributor or people manager?"; and (7) Final Questions about the survey, e.g. "How do you feel about the length of the survey this year?" from 73,268 participants.

8 https://insights.stackoverflow.com/survey/.

3.2 Participants

For our analysis, we selected responses from participants who indicated Germany as their current place of residence, resulting in an initial sample of 5,395 participants. Responses were excluded if participants were under 18 years old or had skipped any of the survey questions essential to the cluster analysis, which resulted in a final sample of 2,837 participants. Missing values were not replaced as the number of missing responses was substantial (46%), and the reduced data set was thought to be still large enough to meet the sample size standards recommended for cluster analysis. Recommendations that depend on the number of clustering variables vary from a minimum of 10-100 times the number of clustering variables [39], which would here correspond to a minimal sample size of 50-500 survey participants.

3.3 Data Preparation and Analysis

Prior to analysis, data were encoded and transformed into new variables where necessary. The original response option "I am a developer by profession" (see 3.1) was encoded as "yes", and all other response options in this question as "no". To reduce redundancy in the cluster analysis, the original twenty-nine individual developer jobs were summarized as higher-order categories, describing the primary focus or context of software development: "Development and programming", "Data and analytics", "Infrastructure and operations", "Management and leadership", "Education and research" or "Student". Response options to multiple choice questions were included in the analysis as separate variables with a dichotomous response format ("yes"/"no"). Ordinal categorical data were numerically encoded. For the cluster analysis, all numerical data were z-standardized and nominal data were encoded using one-hot encoding.

The prepared data set was then subjected to cluster analysis. Cluster analysis is a procedure typically used in market research to identify different consumer types [39]. Here, it was used to identify sub-groups within the German developer community and to characterize them in terms of sociodemographic aspects, use of tools and technology, and their involvement in the Stack Overflow community. Cluster analysis aims to increase the homogeneity within groups while increasing the heterogeneity between groups. As software development plays a crucial role in different areas of industries and academia, we expected to find multiple clusters.

The selection of clustering variables included socioeconomic variables, namely level of education level and yearly income, employment as professional software, coding experience in years, and the development context. To determine the appropriate number of clusters, the elbow method and a dendrogram were used. As k-means clustering can be problematic with nominal data [7], hierarchical clustering was applied using Ward's algorithm and Gower's distance. Gower's distance has the advantage that it combines Manhattan distance for numeric data and Dice distance for nominal data. When computing Gower's distance, the variables "Level of education" and "Developer by profession" were included with double weight to account for their importance. Conversely, the six variables related to "Development context" were weighted with $1/6$ to account for the fact that they were actually response options of the same multiple choice question.

To cross-validate and further explore our clustering results, the same analysis was repeated on another country-based sample from the Stack Overflow data set. The sample for cross-validation was chosen from the list of top ten countries with a focus on appropriate sample size and relative cultural similarity to the German sub-sample. Interpretation of the resulting clusters was finally based on descriptive statistics of the clustering variables and a comparison of the clusters. To test for cluster effects and differences in the clustering variables between clusters, one-way analysis of variance (ANOVA) and pair-wise comparison using Tukey's HSD were employed.

In addition to the analysis of sociodemographic variables, we visualized the use of different programming languages and other software development tools to better interpret the clusters. For programming languages, we looked at differences in use ($\geq 1\%$) for each clusters compared to the total sample to identify languages that were particularly prominent in these clusters and for development environments, frameworks and libraries, version control systems and other developers tools we visualized the relative frequency of use as radar charts to compare profiles of developers from different clusters.

As we were interested in how involved the identified developer sub-groups were in the Stack Overflow community, we analyzed the relationship between coding experience, as one of the defining clustering variables on the one hand, and the subjectively assessed level of community membership (responses ranging from -3 - "No, not at all" to 3 - "Yes, definitely"), the frequency of visiting Stack Overflow (responses ranging from 1 - "Less than once per month or monthly" to 5 - "Multiple times per day") and the frequency of participating in Q&A on Stack Overflow (responses ranging from 0 - "Never" to 5 - "Multiple times per day"), as parameters of the community involvement on the other hand. To determine the strength and direction of the relationship we used Pearson's correlation coefficient. Differences in the community variables between cluster were analyzed using ANOVA and pair-wise comparison tests (Tukey's HSD). As a last measure of community involvement differences, we compared the proportions of developers who indicated to have a Stack Overflow account for each cluster to the total sample using χ^2 test. As the time spent in the community correlates positively with the sense of community [33], we expected to find a positive relationship between coding experience and community membership. Several studies have shown that the sense of community [29] is a driving factor of community engagement, e.g. [35,37]. Other studies have shown that the sense of community, usually discussed in the context of neighborhood communities, also plays a crucial role in virtual communities [1,14]. Therefore, we also hypothesized a positive relationship between community membership, as one aspect of the sense of community, and the frequency of visiting Stack Overflow and participating in Q&A.

Data preparation and analysis were carried out using Python (3.9) and the following libraries: Pandas (2.0.3) [49], NumPy (1.25.1) [17], SciPy (1.11.1) [53], Scikit-learn (1.3.0) [36], Matplotlib (3.7.2) [21] and seaborn (0.13.0) [54].

4 Results

4.1 Sample Characteristics

The final sample consisted of survey responses from 2,837 participants, who mostly identified as men (95%). Half of the participants (50%) were between 25 and 34 years old, and the majority (66%) had a Bachelor's or Master's degree (see Table 1). The Dutch sub-sample, that was used to cross-validate the results of the cluster analysis, had similar characteristics (see Table 2). Similar to the German sample, most participants included in the Dutch sample (n=885) identified as men (93%). Half of the participants (50%) were between 25 and 34 years old, and the majority (75%) had a Bachelor's or Master's degree.

4.2 Cluster Analysis

Elbow method and dendrogram suggested three, four or even five clusters. The four-cluster solution was chosen to obtain clusters as homogeneous as possible without over-fitting (the fifth cluster reduced the within-cluster sum of squares only marginally). The cluster analysis (Ward's algorithm, Gower's distance) showed that the German Stack Overflow developer community could be separated into four characteristic groups (Table 1). Of these four clusters, three clusters consisted of professional developers, Cluster 1, 2, and 3, which were distinguished by a different level of education, ranging from developers mostly likely trained as "Fachinformatiker:in" (IT specialist without a university degree) in Cluster 1 over developers with a Bachelor's degree in Cluster 2 to developers with a Master's degree in Cluster 3, and a working context strongly focusing on "Development and Programming" as well as "Infrastructure and operations". Developers of Cluster 4, on the other hand, had a more diverse educational background and were less specialized regarding the context of software development but had a strong focus on "Education and Research" as well as "Data and Analytics". Similar results were obtained for the Dutch sample (Table 2)

A one-way ANOVA showed a significant difference in programming experience across clusters, $F(3, 2833) = 32.72$, $p < .001$. Developers of Cluster 3 had an on average higher programming experience (see Table 1) than developers of all other clusters ($p < .001$ for comparison with Cluster 1 and Cluster 2 and $p = .006$ for comparison with Cluster 4) and developers of Cluster 4 had significantly more programming experience than those of Cluster 2 ($p = .004$; $p > .05$ for all other comparisons). The income was also found to be significantly different across clusters, $F(3, 2833) = 34.89$, $p <.001$, and developers of Cluster 3 had the highest income ($p < .001$ for all comparisons), while the other clusters did not differ significantly ($p > .05$ for all comparisons). Similarly, there was a big difference in terms of educational level, $F(3, 2833) = 3982.28$, $p < .001$. Developers of Cluster 1 had a significantly ($p < .001$) lower level of education ($M = 1.67$, $SD = 0.68$) than developers of Cluster 2 ($M = 4.00$, $SD = 0.00$), Cluster 3 ($M = 5.10$, $SD = 0.40$) and Cluster 4 ($M = 4.34$, $SD = 1.55$). The level of education was lower for Cluster 2 compared to Cluster 3 and 4 ($p < .001$

Table 1. Sociodemographic characteristics of participants from Germany

	Total	Cluster 1	Cluster 2	Cluster 3	Cluster 4
n	2,837	696	739	1105	297
Developer by profession (%)					
Yes	90	100	100	100	0
No	10	0	0	0	100
Years of coding experience					
M	15.63	14.77	13.60	17.53	15.66
SD	8.93	9.12	7.63	9.01	9.78
Level of education (%)					
Finished school	12	43	0	0	9
College/university without degree	12	45	0	0	10
Associate degree	3	12	0	0	1
Bachelor's degree	29	0	100	3	18
Master's degree	37	0	0	84	40
Doctoral degree	7	0	0	13	22
Context of software development (%)					
Data & analytics	13	6	7	17	29
Development & programming	87	96	95	90	35
Education & research	21	16	14	19	57
Infrastructure & operations	30	32	26	29	39
Management & leadership	16	15	14	15	29
Student	1	2	2	0	4
Income (USD)					
Median	69,318	58,654	65,053	76,783	63,986
IQR	36,259	34,126	29,860	31,995	39,458
Age (%)					
18-24 years	10	22	12	2	6
25-34 years	50	43	60	48	51
35-44 years	29	24	22	36	27
45-54 years	8	7	5	11	10
55-64 years	3	3	2	3	4
>65 years	0	0	0	0	0
Gender (%)					
Man	95	94	95	95	94
Other	2	4	2	2	4
Woman	4	4	5	4	4
Accessibility (%)					
Differences in the ability to hear or see	2	3	2	1	2
Differences in the ability to type, walk or stand	0	0	0	0	0
None of the above	97	96	98	98	98
Psychological or neurological differences (%)					
Affective disorder	10	20	10	6	8
Anxiety disorder	6	11	6	4	5
Neurodiversity	12	19	12	8	12
None of the above	78	65	79	85	78

Table 2. Sociodemographic characteristics of participants from the Netherlands

	Total	Cluster 1	Cluster 2	Cluster 3	Cluster 4
n	885	146	433	244	62
Developer by profession (%)					
Yes	93	100	100	100	0
No	7	0	0	0	100
Years of coding experience					
M	15.45	15.52	14.12	17.25	17.53
SD	9.33	10.24	8.82	9.03	10.26
Level of education (%)					
Finished school	3	26	0	0	0
College/university without degree	13	74	0	0	11
Associate degree	4	0	8	0	5
Bachelor's degree	47	0	92	0	23
Master's degree	28	0	0	91	48
Doctoral degree	4	0	0	9	13
Context of software development (%)					
Data & analytics	15	7	10	21	32
Development & programming	89	98	94	88	39
Education & research	20	15	13	24	56
Infrastructure & operations	27	38	26	23	24
Management & leadership	13	14	10	13	24
Student	1	1	1	0	3
Income (USD)					
Median	66,540	63,036	63,288	78,037	71,985
IQR	46,923	41,770	40,525	48,524	56,125
Age (%)					
18-24 years	11	14	15	3	11
25-34 years	51	53	54	47	36
35-44 years	26	22	21	37	27
45-54 years	9	8	7	9	19
55-64 years	3	4	3	3	5
>65 years	0	0	0	0	2
Gender (%)					
Man	94	98	93	93	95
Other	2	3	1	2	0
Woman	5	1	6	6	5
Accessibility (%)					
Differences in the ability to hear or see	3	3	2	2	9
Differences in the ability to type, walk or stand	0	1	0	0	2
None of the above	96	96	97	97	88
Psychological or neurological differences (%)					
Affective disorder	8	8	10	5	9
Anxiety disorder	9	11	10	6	5
Neurodiversity	23	35	25	16	14
None of the above	69	57	66	79	75

for both comparisons) and also lower for Cluster 4 compared to Cluster 3 ($p <$ .001).

Differences were also found regarding the context of software development. Regarding Cluster 1, a significantly larger proportion of developers worked in the context of "Development and Programming" (χ^2 (1, n = 696) = 38.14, p < .001) and a smaller proportion in "Data and Analytics" (χ^2 (1, n = 696) = 26.06, p < .001) as well as "Education and Research" (χ^2 (1, n = 696) = 9.26, p = .002), but no differences were found for "Infrastructure and Operations" (χ^2 (1, n = 696) = 1.08, p = .298) or "Management and Leadership" (χ^2 (1, n = 696) = 0.74, p = .389) or the proportion of students (χ^2 (1, n = 696) = 1.89, p = .170) compared to the total sample. Developers of Cluster 2 as well worked more often in "Development and Programming" (χ^2 (1, n = 739) = 25.58, p < .001), but less often in "Data and Analytics" (χ^2 (1, n = 739) = 22.84, p < .001). They also worked less often in "Education and Research" (χ^2 (1, n = 739) = 21.77, p < .001), "Infrastructure and Operations" (χ^2 (1, n = 739) = 5.53, p = .019) and "Management and Leadership" (χ^2 (1, n = 739) = 4.10, p = .043). The proportion of students did not differ from the total sample either (χ^2 (1, n = 739) = 0.04, p = .838). Similar to the first two clusters, developers of Cluster 3 had a stronger focus on "Development and Programming" (χ^2 (1, n = 1105) = 10.13, p = .001) and, in contrast to these, worked slightly more often in "Data and Analytics" (χ^2 (1, n = 1105) = 13.66, p < .001) and were less often students (χ^2 (1, n = 1105) = 13.73, p < .001). Cluster 3 did not differ regarding the proportions of developers working in "Education and Research" (χ^2 (1, n = 1105) = 1.68, p = .195), "Infrastructure and Operations" (χ^2 (1, n = 1105) = 0.59, p = .444) or "Management and Leadership" (χ^2 (1, n = 1105) = 0.90, p = .342). In contrast to all other clusters, developers of Cluster 4 worked significantly less frequently in "Development and Programming" (χ^2 (1, n = 297) = 545.04, p < .001). Instead, they worked most frequently in "Data and Analytics" (χ^2 (1, n = 297) = 66.33, p < .001), "Education and Research" (χ^2 (1, n = 297) = 211.07, p < .001), "Infrastructure and Operations" (χ^2 (1, n = 297) = 12.41, p < .001) as well as "Management and Leadership" (χ^2 (1, n = 297) = 37.46, p < .001) and also had a higher proportion of students (χ^2 (1, n = 297) = 19.07, p < .001) than the overall sample.

4.3 Tools and Technology

Socioeconomic differences and differences in the development context went along with differences in tool use (Fig. 1), in particular, for Cluster 4, which was comprised of developers who mostly worked in the context of research and education. For this cluster, we observed a much stronger focus on the programming languages Python, R, and MATLAB compared to other clusters (Fig. 1a). Developers of the first cluster used programming languages that are typical of web development more frequently, such as HTML/CSS, JavaScript, PHP, and TypeScript. While the profiles of the clusters largely overlapped with respect to development environments (Fig. 1b), frameworks (Fig. 1c), and other developer tools (Fig. 1d), developers of Cluster 4 showed some special characteristics: more frequent use of IPython/Jupyter and PyCharm as well as Pandas and NumPy and less frequent use of most other developer tools such as Docker. Regarding version control systems, however, Git proved to be the standard for developers of all clusters 1e).

4.4 Community

Analyzing the relationship between coding experience and variables of community involvement, we found a weak positive relationship between coding experience and membership and participation but a weak negative relationship between coding experience and visiting Stack Overflow (Table 3). A moderate positive relationship was observed between community membership and Stack Overflow visits as well as Stack Overflow visits and participating in Q&A. Community membership and participation in Q&A also correlated positively, supporting previous findings that a stronger sense of community positively impacts community commitment. The same pattern was found when relationships were analyzed for each cluster separately. Direction and strength of the relationships were the same, with the exception of the relationship between coding experience and Stack Overflow visits in Cluster 4. In this sub-group, coding experience and Stack Overflow visits were unrelated (r = -.00, p = .982).

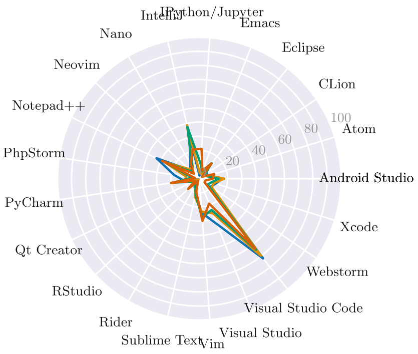
Table 3. Descriptive statistics for variables of community involvement and correlations (Pearson's correlation coefficient) with coding experience

Variable	n	M	SD	1	2	3	4
1. Coding experience	2,837	15.63	8.93	—			
2. Community membership	2,813	0.03	1.17	.09***	—		
3. Community visits	2,831	3.55	1.04	−.07***	.26***	—	
4. Community participation	2,373	1.38	1.04	.16***	.47***	.31***	—

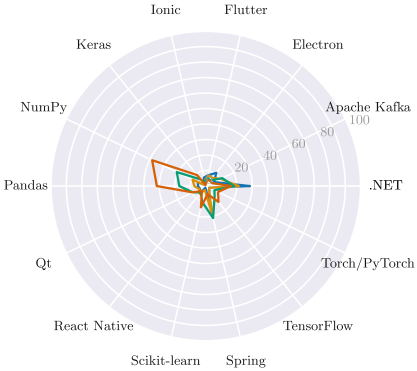
*p < .05, **p < .01, ***p < .001

As both coding experience and community membership proved to be directly related to community engagement, differences in variables of the community involvement between clusters were tested using one-way ANOVA and Tukey's HSD. Results of the ANOVA showed significant differences for all three aspects of community involvement. The average strength of community membership (Cluster 1 - 4: M = -0.11, SD = 1.19; M = 0.04, SD = 1.17; M = 0.13, SD = 1.16; M = -0.03, SD = 1.15) differed between clusters, $F(3, 2809) = 6.08$, $p < .001$, although pair-wise comparison showed that only developers of Cluster 3 considered themselves significantly more strongly as part of the Stack overflow community than developers of Cluster 1 ($p < .001$; $p > .05$ for all other comparisons). The frequency of visiting Stack Overflow (Cluster 1 - 4: M = 3.54, SD = 1.07; M = 3.61, SD = 1.00; M = 3.60, SD = 1.03; M = 3.21, SD = 1.05) differed between clusters as well, $F(3, 2827) = 12.69$, $p < .001$; in particular developers

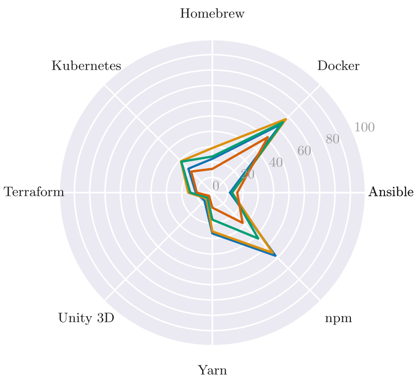
(a) Programming languages



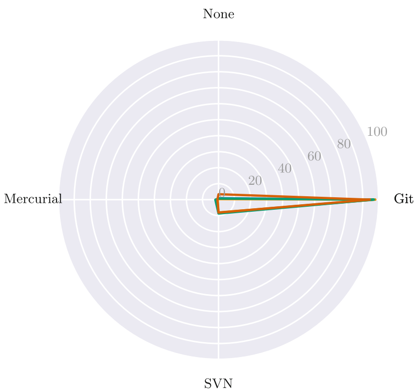
(b) Development environments



(c) Frameworks and libraries



(d) Developer tools



(e) Version control systems

Fig. 1. Comparison of tool use over the past year for four clusters of developers.

of Cluster 4 visited Stack Overflow less often than developers of the other clusters ($p < .001$; $p > .05$ for all other comparisons). Finally, active participation in the Stack Overflow community (Cluster 1 - 4: $M = 1.26$, $SD = 0.96$; $M = 1.35$, $SD = 0.96$; $M = 1.48$, $SD = 1.10$; $M = 1.34$, $SD = 1.06$), that is, asking, answering or voting for questions, was found to be different between clusters, $F(3, 2369) = 6.22$, $p < .001$. Pair-wise comparison showed that developers of Cluster 3 were more engaged in these activities than developers of Cluster 1 ($p < .001$) but not more engaged than developers of Cluster 2 or 4 ($p > .05$ for both comparisons). Comparisons between the other clusters did not reach significance level ($p > .05$). None of the clusters were more or less likely to create a Stack Overflow account (Cluster 1 - 4: 84%, χ^2 $(1, n = 696) = 0.05$, $p = .817$; 84%, χ^2 $(1, n = 737) = 0.00$, $p = .969$; 85%, χ^2 $(1, n = 1105) = 1.13$, $p = .288$; 81%, χ^2 $(1, n = 297) = 2.42$, $p = .119$)

5 Discussion

Cluster analysis is typically used to group objects, persons, or other entities with the aim of identifying groups that share similar characteristics. For instance, in market research, cluster analysis is used to segment customers based on purchasing behaviors, preferences, and demographic factors, enabling more targeted marketing and product strategies. Similarly, here, we were able to identify four sub-groups of developers that could be used to construct the following personas:

(1) **The web developer** who is characterized by moderate programming experience and a distinct specialization in their professional developer career focusing on web development tools. The relatively low educational level is mirrored in lower income rates, suggesting that they might be high school graduates and do not have a professional degree. Nevertheless, they take on programming and development tasks, emphasizing their dedication to acquiring practical skills and experience in the domain.

(2) **The junior developer**, marked by the lowest programming experience, while having a higher educational level than the web developer, is considered to be in the early career phase. They engaged in professional software development and programming with a trend to specialization within their field.

(3) **The senior developer** has the highest programming experience and gets compensated for it more than the other personas. With an above average educational level, they are involved in many different roles, often involving lead functions. For them, the working context is slightly more diverse and requires deeper technological expertise.

(4) **The research developer** represents a niche group characterized by their substantial programming expertise and a distinct inclination towards the research context. They are characterized by a highly diverse range of educational backgrounds, with an average level suggesting a prevalence of advanced educational research degrees. The professional focus here diverges

from explicit software development roles to deeply embedded data analytics ones.

Our results thus underline the significant differences in programming experiences and the early yearly income they receive for their work across the identified clusters, with cluster three standing out for its high levels of both. The educational backgrounds vary strongly. This demonstrates diverse backgrounds and career paths within the German software developer community and is supported by the roles members in each cluster have. With this first glance at the landscape of the German developer community and its structure, we offer valuable insights for employers, educators, and developers themselves. Results emphasize the importance of suitably tailored solutions for workplace design, tool development, and educational and community-building activities. However, with this description of different groups within the community, questions now arise about their influence and connection with requirements for workplace design. As our analysis included only sociodemographic, but no psychometric or behavioral aspects, it also remains unclear whether certain personality traits can be assigned to individual groups and how this could be optimally taken into account in the training and team composition. Besides identifying sub-groups within the community, we also found relationships between coding experience and community involvement among developers in Germany, quantified by their participation in the Stack Overflow community. Findings reveal nuanced dynamics between coding experience, felt community membership, and level of community engagement. Meanwhile, "coding experience" positively correlated with "community membership" "community participation" and "community membership", in turn, with "community participation", indicating that the longer a software developer has been involved in coding, the higher the involvement in the community. Notably, "coding experience" correlated negatively with "community visits" and this relationship did not exist in research software developers. This raises questions about the factors influencing their community engagement, suggesting potential areas for further investigation, such as specific professional needs and alternative support systems. Candidate explanations for this observation could be that the more experienced developers become, the less they need the community knowledge or that, in particular, developers of the research context use other resources. Results also showed significant differences in community involvement across identified clusters where "senior developers" are more engaged in the community compared to "web developers". "Research software developers" visit the community less frequently than the other groups. Results underline the importance of coding experience for a sense of community and engagement. Overall, results highlight a complex interplay between individual experiences and community dynamics. This suggests a need for closer examination of other influencing factors in order to derive implications for community building.

5.1 Limitations

While the present study offers important insights into the diversity of the German software developer community through exploratory cluster analysis, results have to be interpreted with respect to methodological limitations and implications of the data source characteristics. First of all, it should be noted that the study relies on only one community platform and survey data from platform operators. Therefore, the sample is self-selected and biased towards one of many communities within Germany. Additionally, 46% of the initial sample was excluded due to missing responses, which could raise concerns about the potential introduction of non-response bias. The decision to not impute missing values prevented, on the one hand, skewing the data, but on the other hand, this could implicate that the present analysis could have overlooked patterns that might exist within the real world or the complete data set. Additionally, the choice of variables and clustering methods are highly exploratory and subjective. The reliance on pre-selected clustering variables, according to result interpretation, might have been influenced by confirmation bias, inadvertently supporting preconceived notions about the nature of the community. Different choices here might yield different groupings, which influences the study's reproducibility and generalizability of its conclusions. To attempt the latter, we cross-validated findings with a Dutch sub-sample, assuming cultural similarity. However, this additionally might have impacted the transferability. As we analyzed data from a survey that was not designed to address specific research questions and objectives, data on potentially important (human) factors such as attitudes, personality and preferences were not available and could, therefore, not be included in the analysis. Moreover, quality of the available data could not be assessed due to the lack of controls such as seriousness checks. Finally, single-item operationalizations of complex psychological constructs as used here might further affected the validity of our results. Nevertheless, the Stack Overlflow Developer Survey data set provided an economical way to explore the German developer community in a large-scale sample.

5.2 Conclusion and Future Work

In conclusion, the present work provides some novel insights into the diversity in the German developer community, highlighting the significance of a strong sense of community. The findings underscore the importance of fostering community engagement and taking further steps to analyze diversity factors and their relation to software developer needs and requirements towards different aspects of community building and workplace design. Future research could investigate the predictors and outcomes of a strong sense of community within the software developer community, not only in large-scale online communities but also in level workplace contexts. By understanding these dynamics, organizations could benefit in terms of knowledge management strategies, enhancing workplace design, or community-building and collaboration strategies. Providing a cluster analysis based on selected variables, including demographic variables beyond education and employment status, might reveal additional insights, especially into the diversity of the developer community.

Acknowledgement. We gratefully acknowledge the HIFIS (Helmholtz Federated IT Services) team for their generous funding and support of our research. For more information on their services, please visit https://www.hifis.net/. The authors also wish to thank Stack Overflow for making the data set publicly available and the numerous respondents for participating in the survey.

Disclosure of Interests. The authors have no competing interests.

References

1. Abfalter, D., Zaglia, M.E., Müller, J., Kraler, F.: Relevanz und messung von sense of community im virtuellen kontext. In: GI-Jahrestagung (2011). https://api.semanticscholar.org/CorpusID:25722258
2. Ahmed, T., Srivastava, A.: Understanding and evaluating the behavior of technical users. A study of developer interaction at stackoverflow. Human-centric Comput. Inf. Sci. **7**, 1–18 (2017). https://doi.org/10.1186/s13673-017-0091-8
3. Aichroth, P., et al.: Wertschöpfung durch software in deutschland (2021)
4. Anderson, A., Huttenlocher, D., Kleinberg, J., Leskovec, J.: Discovering value from community activity on focused question answering sites: a case study of stack overflow. In: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2012, pp. 850–858. Association for Computing Machinery, New York (2012). https://doi.org/10.1145/2339530.2339665
5. Bagnall, P.: Using Personas Effectively, pp. 221–221 (2007). https://doi.org/10.14236/EWIC/HCI2007.84
6. Bostancioglu, A.: Factors affecting English as a foreign language teachers' participation in online communities of practice: the case of webheads in action. Int. J. Lang. Educ. **4**, 20–35 (2016). https://doi.org/10.18298/IJLET.1651
7. Bruce, P., Bruce, A., Gedeck, P.: Practical Statistics for Data Scientists: 50+ Essential Concepts Using R and Python. O'Reilly Media, Inc. (2020)
8. Cajander, Å., Larusdottir, M., Eriksson, E., Nauwerck, G.: Contextual personas as a method for understanding digital work environments. In: Abdelnour Nocera, J., Barricelli, B.R., Lopes, A., Campos, P., Clemmensen, T. (eds.) HWID 2015. IAICT, vol. 468, pp. 141–152. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-27048-7_10
9. Cameron, C., Wasacase, T.: Community-driven health impact assessment and asset-based community development: an innovate path to community well-being. In: Phillips, R., Wong, C. (eds.) Handbook of Community Well-Being Research. IHQ, pp. 239–259. Springer, Dordrecht (2017). https://doi.org/10.1007/978-94-024-0878-2_13
10. Courage, C., Baxter, K.: Understanding Your Users: A Practical Guide to User Requirements Methods, Tools, and Techniques. Gulf Professional Publishing (2005)
11. Deal, B., Pan, H., Pallathucheril, V., Fulton, G.: Urban resilience and planning support systems: the need for sentience. J. Urban Technol. **24**, 29 – 45 (2017). https://doi.org/10.1080/10630732.2017.1285018
12. Dotan, A., Maiden, N., Lichtner, V., Germanovich, L.: Designing with only four people in mind? – A case study of using personas to redesign a work-integrated learning support system. In: Gross, T., et al. (eds.) INTERACT 2009. LNCS, vol. 5727, pp. 497–509. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-03658-3_54

13. Faisal, M., Issa, G.F., Ayub, I., Asadullah, M., Joiya, U.N., Iqbal, M.: How automate requirements engineering system effects and support requirement engineering. In: 2022 International Conference on Business Analytics for Technology and Security (ICBATS), pp. 1–3 (2022). https://doi.org/10.1109/ICBATS54253.2022.9758997
14. González-Anta, B., Orengo, V., Zornoza, A.M., Peñarroja, V., Martínez-Tur, V.: Understanding the sense of community and continuance intention in virtual communities: the role of commitment and type of community. Soc. Sci. Comput. Rev. **39**, 335 – 352 (2019). https://api.semanticscholar.org/CorpusID:199004874
15. Günzler, P.: Protokoll der fachgruppensitzung der fachgruppe frauen und informatik in naumburg (2023)
16. Gupta, A., Vardhan, H., Varshney, S., Saxena, S., Singh, S., Agarwal, N.: Kconnect: the design and development of versatile web portal for enhanced collaboration and communication. EAI Endorsed Trans. Scalable Inf. Syst. **11**(2), 1–7 (2024)
17. Harris, C.R., et al.: Array programming with NumPy. Nature **585**(7825), 357–362 (2020). https://doi.org/10.1038/s41586-020-2649-2
18. Heller, B., Marschner, E., Rosenfeld, E., Heer, J.: Visualizing collaboration and influence in the open-source software community. In: Proceedings of the 8th Working Conference on Mining Software Repositories, pp. 223–226 (2011). https://doi.org/10.1145/1985441.1985476
19. Hofmann, B., Wulf, V.: Building communities among software engineers: the ViSEK approach to intra- and inter-organizational learning. In: Henninger, S., Maurer, F. (eds.) LSO 2002. LNCS, vol. 2640, pp. 25–33. Springer, Heidelberg (2003). https://doi.org/10.1007/978-3-540-40052-3_4
20. Horwood, C., Youngleson, M.S., Moses, E., Stern, A.F., Barker, P.: Using adapted quality-improvement approaches to strengthen community-based health systems and improve care in high HIV-burden sub-saharan African countries. AIDS **29**, S155–S164 (2015). https://doi.org/10.1097/QAD.0000000000000716
21. Hunter, J.D.: Matplotlib: a 2D graphics environment. Comput. Sci. Eng. **9**(3), 90–95 (2007). https://doi.org/10.1109/MCSE.2007.55
22. Ingram, C., Drachen, A.: How software practitioners use informal local meetups to share software engineering knowledge. In: Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering, pp. 161–173 (2020). https://doi.org/10.1145/3377811.3380333
23. Ismail, A., Sulaiman, S.: A model for knowledge portal to support communities of practice. In: 2011 Malaysian Conference in Software Engineering, pp. 451–457. Johor Bahru, Malaysia (2011). https://doi.org/10.1109/MySEC.2011.6140715
24. Kalliamvakou, E., Damian, D., Singer, L., German, D.M.: The code-centric collaboration perspective: Evidence from github. The Code-Centric Collaboration Perspective: Evidence from Github, Technical Report DCS-352-IR, University of Victoria, p. 17 (2014)
25. Khazaei, A., Elliot, S., Joppe, M.: Fringe stakeholder engagement in protected area tourism planning: inviting immigrants to the sustainability conversation. J. Sustain. Tourism **25**, 1877 – 1894 (2017). https://doi.org/10.1080/09669582.2017.1314485
26. Kraljić, T., Kraljić, A.: ERP implementation: requirements engineering for ERP product customization. In: Themistocleous, M., Papadaki, M. (eds.) EMCIS 2019. LNBIP, vol. 381, pp. 567–581. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-44322-1_42
27. Maida, C., Beck, S.: Towards communities of practice in global sustainability. Anthropology Action **23**, 1–5 (2016). https://doi.org/10.3167/AIA.2016.230101

28. Matthews, T., Whittaker, S., Moran, T., Yang, M.: Collaboration personas: a framework for understanding designing collaborative workplace tools. In: Workshop "Collective Intelligence In Organizations: Toward a Research Agenda." at Computer Supported Cooperative Work (CSCW). Citeseer (2010)
29. McMillan, D.W., Chavis, D.M.: Sense of community: a definition and theory. J. Community Psychol. **14**(1), 6–23 (1986). https://doi.org/10.1002/1520-6629(198601)14:1
30. Miaskiewicz, T., Kozar, K.: Personas and user-centered design: how can personas benefit product design processes? Des. Stud. **32**, 417–430 (2011). https://doi.org/10.1016/J.DESTUD.2011.03.003
31. Middleton, J., et al.: Which contributions predict whether developers are accepted into GitHub teams. In: Proceedings of the 15th International Conference on Mining Software Repositories, pp. 403–413 (2018)
32. Moutidis, I., Williams, H.T.: Community evolution on stack overflow. PLoS ONE **16**(6), e0253010 (2021). https://doi.org/10.1371/journal.pone.0253010
33. Oishi, S., et al.: The socioecological model of procommunity action: the benefits of residential stability. J. Personality Soc. Psychol. **93**(5), 831–844 (2007). https://doi.org/10.1037/0022-3514.93.5.831
34. Oliveira, L., Bradley, C., Birrell, S., Tinworth, N., Davies, A., Cain, R.: Using passenger personas to design technological innovation for the rail industry. In: Kováčiková, T., Buzna, Ľ, Pourhashem, G., Lugano, G., Cornet, Y., Lugano, N. (eds.) INTSYS 2017. LNICST, vol. 222, pp. 67–75. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-93710-6_8
35. Omoto, A.M., Snyder, M.: Influences of psychological sense of community on voluntary helping and prosocial action. In: Stürmer, S., Snyder, M. (eds.) The Psychology of Prosocial Behavior, pp. 223–243. Wiley (2009). https://doi.org/10.1002/9781444307948.ch12, https://onlinelibrary.wiley.com/doi/10.1002/9781444307948.ch12
36. Pedregosa, F., et al.: Scikit-learn: machine learning in Python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)
37. Perkins, D.D., Long, D.A.: Neighborhood sense of community and social capital. In: Snyder, C.R., Fisher, A.T., Sonn, C.C., Bishop, B.J. (eds.) Psychological Sense of Community, pp. 291–318. Springer US, Boston, MA (2002). https://doi.org/10.1007/978-1-4615-0719-2_15, http://link.springer.com/10.1007/978-1-4615-0719-2_15, series Title: The Plenum Series in Social/Clinical Psychology
38. Reed, M., Godmaire, H., Abernethy, P., Guertin, M.: Building a community of practice for sustainability: strengthening learning and collective action of Canadian biosphere reserves through a national partnership. J. Environ. Manage. **145**, 230–9 (2014). https://doi.org/10.1016/j.jenvman.2014.06.030
39. Sarstedt, M., Mooi, E.: Cluster analysis. In: A Concise Guide to Market Research. STBE, pp. 301–354. Springer, Heidelberg (2019). https://doi.org/10.1007/978-3-662-56707-4_9
40. Schäfer, K., et al.: Datenbasierte personas älterer endbenutzer für die zielgruppenspezifische entwicklung innovativer informations-und kommunikationssysteme im gesundheitssektor [data-based personas of older end users for the development of innovative information and communication systems in the health sector for specific target groups]. Zeitschrift für Arbeitswissenschaft **73**(2), 177–192 (2019). https://doi.org/10.1007/s41449-019-00150-5
41. Schäfer, K., et al.: Survey-based personas for a target-group-specific consideration of elderly end users of information and communication systems in the German

health-care sector. Int. J. Med. Informatics **132**, 103924 (2019). https://doi.org/10.1016/j.ijmedinf.2019.07.003
42. Schelhowe, H., et al.: Medienbildung für die persönlichkeitsentwicklung, für die gesellschaftliche teilhabe und für die entwicklung von ausbildungs-und erwerbsfähigkeit (2009)
43. Schlauch, T., Haupt, C.: Using knowledge exchange workshops to analyze the DLR software engineering community. In: 9th International Workshop on Sustainable Software for Science: Practice and Experiences (WSSSPE6.1) (2018). https://doi.org/10.5281/zenodo.1446474
44. Schlauch, T., Haupt, C., Meinel, M., Schreiber, A.: Analytics and insights about cultivating a software engineering community at DLR. In: 2019 IEEE Aerospace Conference, pp. 1–8. IEEE (2019). https://doi.org/10.1109/AERO.2019.8741902
45. Schmitz, C.L., Stinson, C., James, C.D.: Community and environmental sustainability. Critical Social Work **11**(3), 83–100 (2019). https://doi.org/10.22329/CSW.V11I3.5834
46. Schwarz, R., Hellmig, L., Friedrich, S.: Informatik-monitor (2021)
47. Sharon, D., Anderson, T.: Toolbox: a complete software engineering environment; top drawer. IEEE Softw. **14**, 123–127 (1997). https://doi.org/10.1109/52.582983
48. Srba, I., Bielikova, M.: Why is stack overflow failing? preserving sustainability in community question answering. IEEE Softw. **33**(4), 80–89 (2016). https://doi.org/10.1109/MS.2016.34
49. pandas development team, T.: Pandas-dev/pandas: Pandas (2023). https://doi.org/10.5281/zenodo.8092754
50. Trinkenreich, B., Wiese, I., Sarma, A., Gerosa, M., Steinmacher, I.: Women's participation in open source software: a survey of the literature. ACM Trans. Softw. Eng. Methodol. (TOSEM) **31**(4), 1–37 (2022). https://doi.org/10.1145/3510460
51. Vasilescu, B., Filkov, V., Serebrenik, A.: Stackoverflow and GitHub: associations between software development and crowdsourced knowledge. In: 2013 International Conference on Social Computing, pp. 188–195 (2013). https://doi.org/10.1109/SocialCom.2013.35
52. Verma, A.: Critical review of literature of the impact of workforce diversity (specifically age, gender, and ethnic diversity) on organizational competitiveness. Asian J. Manage. **11**, 125–130 (2020). https://doi.org/10.5958/2321-5763.2020.00020.7
53. Virtanen, P., et al.: SciPy 1.0 contributors: SciPy 1.0: fundamental algorithms for scientific computing in Python. Nat. Methods **17**, 261–272 (2020). https://doi.org/10.1038/s41592-019-0686-2
54. Waskom, M.L.: seaborn: statistical data visualization. J. Open Source Softw. **6**(60), 3021 (2021). https://doi.org/10.21105/joss.03021
55. Wen, S.-F., Kianpour, M., Katt, B.: Security knowledge management in open source software communities. In: Lanet, J.-L., Toma, C. (eds.) SECITC 2018. LNCS, vol. 11359, pp. 53–70. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-12942-2_6