“Get Me In The Groove”: A Mixed Methods Study on Supporting ADHD Professional Programmers

Kaia Newman*, Sarah Snay†, Madeline Endres‡, Manasvi Parikh†, Andrew Begel*
*Carnegie Mellon University, Pittsburgh, PA, USA
†University of Michigan, Ann Arbor, MI, USA
‡University of Massachusetts, Amherst, Massachusetts, USA
{klnewman, abegel}@andrew.cmu.edu, {ssnay, manasvi}@umich.edu, mendres@umass.edu

***Abstract*—Understanding the work styles of diverse programmers can help build inclusive workplaces, enabling all software engineers to excel. An estimated 10.6% of programmers have *Attention Deficit Hyperactivity Disorder* (ADHD), a condition characterized by differences in attention and working memory. Prior work has just begun to explore the impact of ADHD on software development, finding that inadequate support may negatively impact team productivity and employment. This prevents software organizations from benefiting from ADHD-related strengths. To investigate these impacts, we conducted a two-phase mixed methods study. First, we qualitatively analyzed 99 threads (1,658 posts and comments) from `r/ADHD_Programmers`, the largest public forum dedicated to the ADHD programmer community. We constructed a mapping that reveals how ADHD programmers apply personal strategies and organizational accommodations to address software task-specific challenges. Second, we conducted a large-scale survey of 239 ADHD and 254 non-ADHD professional programmers to validate how our qualitative data generalize to the worldwide developer population. Our results show that ADHD programmers are 1.8 to 4.4 times more likely to struggle more frequently than neurotypical developers with all challenges we consider, but especially with time management and design. Our findings have implications for inclusive and effective tool- and policy-building in software workplaces and motivate further research into the experiences of ADHD programmers.**

***Index Terms*—Neurodiversity, ADHD, Software Engineering, Accommodations**

I. Introduction

Attention Deficit Hyperactivity Disorder (ADHD), a condition involving differences in attention and working memory, has been linked to strengths and challenges such as cognitive dynamism [81] and difficulty finishing work efficiently [29]. ADHD is of increasing relevance in software engineering in terms of how the community works to build effective, diverse workplaces and support marginalized groups [31], [49], [59]. Many software developers have ADHD; **10.6% of respondents** in the 2022 Stack Overflow Developer Survey reported having “a concentration and/or memory disorder (e.g., ADHD, etc.)" [86]. ADHD developers have under-addressed challenges compared to their neurotypical peers: common software workplace tasks or tools, such as virtual meeting calls, open office plans, or pull request notifications, can be ill-suited to neurodivergent (e.g., ADHD) programmers’ focus, productivity, or wellbeing [20], [49], [59]. When unsupported, similar challenges in other fields lead to negative outcomes such as underperformance [43] or frequent job switching [60].

Unfortunately, support for ADHD employees is often insufficient [43]; accommodations are rarely requested (even if needed) [4] and difficult to obtain [67]. In one survey, 9 out of 59 (15.5%) neurodivergent professional programmers desired accommodations, but only 1 requested them [59]. Asking for accommodations can be risky: legal protections against discrimination may not guard against more insidious inequities, such as implicit bias or firing employees for nominally-legitimate reasons [22]. In addition, accommodations for particular needs can violate privacy due to their specificity [65], making information about them and how to obtain them harder to navigate. Specific accommodations are usually created from an understanding of barriers and the strategies used to overcome them [36].

There is not yet a complete understanding of the associations between ADHD developer strategies, accommodations, and challenges, how they impact specific software tasks, or how they generalize. This knowledge would allow companies to develop and deploy targeted interventions for software tasks. Although there has been recent growth in researching neurodivergent programmers, prior work mostly focuses on other groups (e.g., autism [2] or dyslexia [53]), does not separate out ADHD software engineers as their own participant group [59], or does not identify software task-specific challenges or relationships between them and support options [49].

Understanding the challenges and strategies of ADHD developers may inform interventions that better support their needs without requiring disclosure. Studying the problem-solving strategies of developers is valuable for tool building [46], and it may also help policy building: GenderMag, a framework for understanding how gender-diverse users interact with software, has helped companies identify and fix inclusiveness issues [13], [61]. Supporting neurodiverse programmers may also lead to organizational-level productivity gains; diverse teaming across multiple attributes has been shown to increase productivity in software contexts [90]. Due to the efficacy of diverse software teams and the prevalence and potential strengths of ADHD software engineers, understanding how to support their unique work styles is merited.

An effective investigation into the experiences of ADHD programmers would both uncover developer experiences from intersectional backgrounds and also lead to insights that generalize to a broad range of developers. We thus carried out

a two-phase mixed methods study. We first qualitatively analyzed data from the subreddit `r/ADHD_Programmers` to gather detailed, subjective experiences. The anonymity of online forums permits relatively safe community-building and disclosure around sensitive topics [68]. We then conducted a large-scale survey of professional programmers to see if these experiences generalize to programmers with and without ADHD. In this work, we focus on programmers' workplace challenges and strengths, personal strategies, and organizational accommodations. Our contributions are:

- A qualitative analysis of 99 posts and 1,559 comments from the forum `r/ADHD_Programmers` (∼160,000 words).
- A survey of 493 professional programmers of varying self-reported neurotypes, including 239 with ADHD.
- A mapping between ADHD programmers' workplace challenges, and the accommodations and strategies they use. This includes the first breakdown by software task and the first academic investigation into existing accommodations for ADHD software engineers.
- We find that while challenges impact ADHD programmers at significantly higher rates, they also impact non-ADHD programmers. As a result, organizational support may lead to generalizable productivity gains.

II. Background

Neurodiversity describes the range of variation in human cognition [64]. The majority cognitive profile is called *neurotypical*; minority neurotypes, including ADHD and autism, are *neurodivergent*. Some minority neurotypes co-occur, e.g., 40% of autistic people also have ADHD [77]. ADHD is present in 5–7% of the global population [91].

People with ADHD have documented strengths including hyperfocus ("episode[s] of long-lasting, highly focused attention" [39]), entrepreneurship [58], and creativity [9]. These strengths have been measured objectively: not only did people with ADHD report significantly more creative achievements in the real world, but they also scored higher on a validated test of divergent thinking when competing with others (with a medium-to-large effect size) [9], [17]. However, people with ADHD struggle disproportionately with challenges, such as distractions, organization, and job loss, which can prevent them from leveraging their strengths in the workplace [5]. For example, one study found that ADHD adults experienced significantly more job-switching than neurotypical adults [60]. Another found that ADHD adults were affected significantly more by distractors during a letter search task [28].

We follow the *social model of disability* [82], which posits that difficulties faced by disabled people, such as those with ADHD, are impacted by social and environmental barriers. To address difficulties, we advocate for removal of these barriers. We now overview existing work on challenges faced by ADHD employees and support in a non-software context.

Research has found that ADHD-related challenges can be reduced by environmental, managerial, and behavioral practices [1], [15], [36], [76], [78]. This includes both personal strategies and organizational accommodations. As an example of the former, one study found that ADHD participants used routines and checklists to aid with distractions, time management, and organization [15]. Another study described strategies such as choosing environments that were best suited to one's ADHD symptoms and using timers [45].

At the organizational level, support helps mitigate distractions. This includes providing a private work space, headphones, or flexible schedules so that employees can choose quieter times [1]. Incentivizing task completion and delegating tedious work can help support ADHD employees [1]. To mitigate for difficulties with communication and time management, research suggests that managers communicate job tasks clearly, including defining timelines and creating a list of job-related responsibilities [76]. Accommodations for ADHD have been shown to be effective in non-software contexts: for example, one review found that providing a list of tasks to choose from rather than assigning tasks directly increased engagement for ADHD students in multiple studies [36].

The intersection of ADHD and software is less studied. Liebel *et al.* conducted a case study, interviewing 19 ADHD professional developers [49]. For a comparison, see §VIII.

III. Research Questions and Methodology

We carried out a two-phase mixed methods study of ADHD programmers. We qualitatively analyzed posts from `r/ADHD_Programmers` and then checked if our findings generalize via a survey of 493 professional programmers. Developed using previous literature and initial readings of this subreddit, we directed our analysis around four research questions:

RQ1: What challenges do ADHD software engineers have, and which software tasks are impacted?

RQ2: What strategies and accommodations do ADHD software engineers use to address challenges at work?

RQ3: Which strategies or accommodations may help with which work challenges?

RQ4: Do non-ADHD programmers also think these strategies and accommodations would be helpful?

We now overview each phase's methodology. Supporting material (including our full qualitative codebook, survey instrument, and analysis scripts) are in our replication package.[1]

A. Phase 1: Qualitative Analysis of Archival Data

Dataset. We desire a nuanced understanding of the ADHD programmer experience in a natural setting. We thus analyzed posts and comments from `r/ADHD_Programmers`, a public forum where ADHD programmers engage in community discussion. As of August 2024, this forum has 61,000 members, making it in the top 2% in size on Reddit. To the best of our knowledge, we are the first to analyze this forum academically and it is the largest public community of ADHD programmers.

We collected posts in late 2022 using the Pushshift API, which is commonly used to create qualitative datasets [6]. We

[1]Our replication package, including our final codebook, is available at https://github.com/kaianew/GetMeInTheGroove_ICSE2025.

TABLE I. % job-related `r/ADHD_Programmers` posts by category, and the 99 analyzed posts. Posts can be in multiple categories.

Post Content	% Posts	Document IDs in Analysis
Challenges	46%	1,3–8,11,13–18,21–24,26–27,29–31,33,35,37–38,41,44, 47–52,57–58,60,64–66,72,74,77–78,80–85,87,89–91,94,97–99
Strategies	44%	3–4,6–10,12–18,20–24,26–32,35,37,41–44,47–51,55,57–58,62,64–67,70,72,74–75,77,80–82,84–85,87,89,91–95,99
Disclosure	9.1%	1–2,5,7,19–20,32–34,37,39–41,45–47,54,59–61,63,67–69, 73,76,78,83,86,94,96,98
Accommodations	8.6%	1,4–5,7–10,13,25,28,33,36–38,40–41,43,45–47,49,53,56, 63,68,71,73,76–77,79,88,91,94,96,98

scraped using RQ-related keywords such as career roles (e.g., "employee"), parts of the software process (e.g., "debugging"), and ADHD-related vocabulary (e.g., "hyperfocus"). This led to 2,037 posts, of which 881 were software-job-related (via manual annotation by the authors). We defined software-job-related using three intersecting criteria: employment-related words in a software context (e.g., job, boss, Agile), phases in the job cycle (e.g., interviewing, job searching, being fired), or posts soliciting comments about software employment.

The first four authors categorized these job-related posts into four non-mutually-exclusive categories: challenges, accommodations, disclosure, and strategies. Of these, challenges and strategies were the most common (see Table I). We focus this work on challenges, strategies, and accommodations, as we found these topics were relevant for software engineering-specific intervention. We intend to publish our findings on disclosure in another venue; that paper *will not* include challenges, strategies, or accommodations. We used separate analyses to model disclosure motivations and outcomes. In addition, the papers differ in methodology and data used.

We chose 15 posts spanning all non-mutually-exclusive categories (challenges, accommodations, strategies, disclosure) to facilitate initial codebook development. The remaining 84 posts were sampled from the 881 software-job-related posts in a stratified manner, i.e., 21 random posts from each category, since the categories were imbalanced in the overall sample (Table I). This follows best practice [40], and helps ensure that our codebook would generalize to a random sample. We reached thematic saturation (i.e., we were not discovering new codes) after 33 posts, lending confidence that our sample size was sufficient. We performed an in-depth qualitative analysis of all 99 job-related posts and their 1,559 comment threads (∼160,000 words: max 66, avg. 16 comments per post).

Analysis. Our thematic analysis followed Deterding and Waters's twenty-first century approach [23]. To build and apply our codebook, we used a three-pass process: one to identify relevant quotes via *semantic unitization* [14], one to apply top-level codes, and one to fully analyze the data. Posts had passes by at least two authors. We applied 12,891 code instances to 3,555 quotes using ATLAS.ti, a qualitative analysis software.

When building the codebook, authors coded the same documents independently and met weekly to resolve conflicts and refine definitions in a process of *negotiated agreement* [14]. When our codebook did not change for three meetings in a row (i.e., reached saturation [30]), the authors started annotating posts alone, discussing edge cases. We achieved high top-level inter-rater reliability (IRR) (Krippendorff's $\alpha = 0.75$ [37]), lending confidence to our categorical consistency. Following best HCI research practice [55], we did not calculate IRR for lower-level codes (despite using negotiated agreement); we use codes to discover themes rather than as our final product.

TABLE II. Demographics overview of our survey population.

Gender / Neurotype	ADHD	Other ND	NT	Total
Man	180 (36.5%)	52 (10.6%)	178 (36.1%)	410 (83.3%)
Woman	29 (5.9%)	4 (0.8%)	13 (2.6%)	46 (9.3%)
Nonbinary	25 (5.1%)	1 (0.2%)	3 (0.6%)	29 (5.9%)
Other/NA	5 (1.0%)	3 (0.6%)	0	8 (1.6%)
Total	239 (48.5%)	60 (12.2%)	194 (39.4%)	**493**

	Min	Max	Median
Age (in Years)	18	74	32
Full Years of Professional Experience	0	48	9

Final Codebook. We identified 168 lower-level codes organized into 11 top-level groups, including WORKCHALLENGE, COPINGMECH (i.e., a personal strategy a developer uses to mitigate challenges), WORKSTRENGTH, and ACCOMMODATION. We define an ACCOMMODATION as any support that needs approval from someone higher up in an organization. *This is explicitly not a legal definition*; commenters used various colloquial definitions and we wanted to capture all support ADHD developers may desire or receive.

Axial Coding. We mapped strategies and accommodations to challenges using axial coding, which involves reassembling codes into abstracted categories to develop relationships between concepts [80]. The first three authors organized codes into a mapping based on challenges, abstracted codes into broader categories, and ensured relationships were grounded in the original coding via code co-occurrence (see Figure 1).

B. Phase 2: Quantitative Analysis of a Large-Scale Survey

Survey Design. To triangulate, validate, and expand on our qualitative results, we designed a 20-minute online survey. We made 5-item Likert-style questions derived directly from our main qualitative themes (e.g., work challenges, strategies, accommodations, and strengths). We consulted with HCI and neurodiversity outreach experts to ensure the quality and sensitivity of our survey. We did not include all qualitative codes or relevant software tasks as it would have made the survey prohibitively long. Instead, we prioritized common themes and concepts. While we have 13 software tasks in our codebook, *we only have 6 in our survey:* code review, debugging, brainstorming, meetings, design, and documentation.

Though we included ADHD-specific questions, we designed our survey to be completable by people of all neurotypes to better contextualize the ADHD experience. We also included a validated assessment of positive mental health [50]; we theorized that mental health may moderate challenge frequency.

Survey Recruitment. We primarily recruited via public GitHub emails. We used the GitHub REST API[2] to identify

[2]https://docs.github.com/en/rest?apiVersion=2022-11-28

the top 25 contributors from the top 100 repositories across 30 languages. After duplicate filtering, we had 17,202 emails.

Our broad recruitment allowed us to reach ADHD programmers, as they make up a minority of developers. We sent emails in batches of ∼5,000 per week to ensure quick response to comments or concerns. GitHub recruitment is well-established in software engineering surveys of professional developers [26], [38], [47], [48], [57]. We discussed the ethics of recruiting via public GitHub emails when choosing this strategy, especially around a potential violation of their terms of service. Following prior arguments [34], [57], we consider the principles of beneficence and respect for persons outlined in the Belmont report [83], and believe that on balance, our work's potential to benefit inclusive software development through insights on neurodiverse programmers outweighs the low costs to GitHub users (e.g., an unwanted email). During the first 1.5 weeks of recruitment, we advertised using posters in three US metro areas, on an email list of programmers interested in our research, and by word of mouth. These approaches resulted in a low yield (n = 26). We considered other strategies, such as MTurk or Prolific, but ultimately decided on GitHub because of data quality concerns [19], [74].

We offered an optional drawing for $120 (or an equivalent amount proportional to the recipient's minimum wage). 502 participants submitted our survey (out of 821 who started, a 61% completion rate). Through self-consistency filters and removing two responses with hate speech, we had 493 valid responses. We recruited from May 16 to June 18, 2024.

Analysis. We organized our statistical analysis around three groups: those with ADHD, those who are neurodivergent but do not have ADHD (Other ND), and neurotypicals (NT). For Other ND, we had a narrow definition, excluding conditions not always considered neurodivergent (e.g., depression, tinnitus). We performed all statistics in an R Notebook [72].

To analyze high-level differences between neurotypes, we used the χ^2 test of independence, a non-parametric test appropriate for categorical data [56]. However, for Likert-type questions on challenge or strength frequency, we theorized there may be confounds beyond neurotype. We use *ordinal logistic regression* (OLM) to determine whether changing a predictor variable significantly affects the response [54]. We use OLMs rather than *linear regression* because Likert-type items are ordered. Though it is appropriate to treat Likert-type data as interval data in some cases, we treated them as ordinal because we do not aggregate responses and responses are not numerical [35]. OLMs rely on the *proportional odds assumption* in which model coefficients must be consistent across the different levels of the response variable [10]. We tested for this assumption to ensure the validity of our models. All model results reported in this work satisfy this assumption. We used the R packages MASS [75] (`polr`) and BRANT [79].

We chose predictors based on related work and our qualitative analysis. For challenge models, we used neurotype, years of experience, positive mental health, work-from-home status, and reported task competency. For strengths, we used neurotype, years of experience, and positive mental health.

We used a significance threshold of $p < 0.05$. As we generated p-values for dozens of models, we used the Benjamini-Hochberg (BH) adjustment to correct for multiple comparisons within research questions [7] and avoided trying factors in models which have no backing from theory. All results reported in this work survived correction unless otherwise noted.

IV. Survey Population Contextualization

Table II contains our survey demographics. All respondents were at least 18 and had worked at a job that required software development. The most common job titles in our sample were "Software Engineer" or "Software Developer" (60%). Participants were from 58 countries across seven regions: North America ($n = 202$), Europe & Central Asia (201), East Asia & Pacific (48), Latin America (22), South Asia (12), Middle East & North Africa (6), and Sub-Saharan Africa (1).

For self-reported neurotype, 239 (48.5%) had ADHD, 60 (12.2%) were neurodivergent without ADHD (Other ND, 73% autistic), and 194 (39.4%) were neurotypical (NT). 99 ADHD participants were also autistic. Our findings are intersectional as we combine all ADHD participants in one group (see §VII).

ADHD diagnosis prevalence varies by culture, socioeconomic status, and gender, and many are undiagnosed [3]. We thus combined diagnosed and self-diagnosed ADHD participants: 112 (47%) were diagnosed, while 127 (53%) were self-diagnosed. We include an analysis of challenges and strengths of diagnosed programmers in our replication package. Diagnostic status does not impact our high-level findings. However, 3.6% of statistical tests with significant differences in the diagnosed group were not significant for the combined group. Also, odds ratios (i.e., effect sizes) in this paper are generally lower than if we had only considered diagnosed programmers.

V. Results: Challenges and Solutions

We identified and validated 5 challenge types that are faced disproportionately by ADHD developers. These challenge types are *Challenges Involving Cognition*, *Challenges Involving Time*, *Challenges Involving Distractions*, *Challenges Involving Communication*, and *Challenges Involving Alignment*. We identified each challenge in our qualitative analysis of `r/ADHD_Programmers` and then validated that they impact ADHD developers disproportionately via our survey. In Fig. 1, **we propose a mapping between these challenges and relevant personal strategies and organizational accommodations** (as determined by our axial coding, see §III-A). We focus on challenges related to work outcomes. We leave an investigation of mental health challenges to future work.

In the rest of this section, we address RQs 1–3 (see §III). We first overview each challenge and discuss which software tasks are most impacted. Table III summarizes our qualitative and quantitative results by challenge. Some tasks identified qualitatively were not included in the survey, and were thus not possible to validate (see §III-B). We then detail strategies and accommodations used by ADHD programmers.

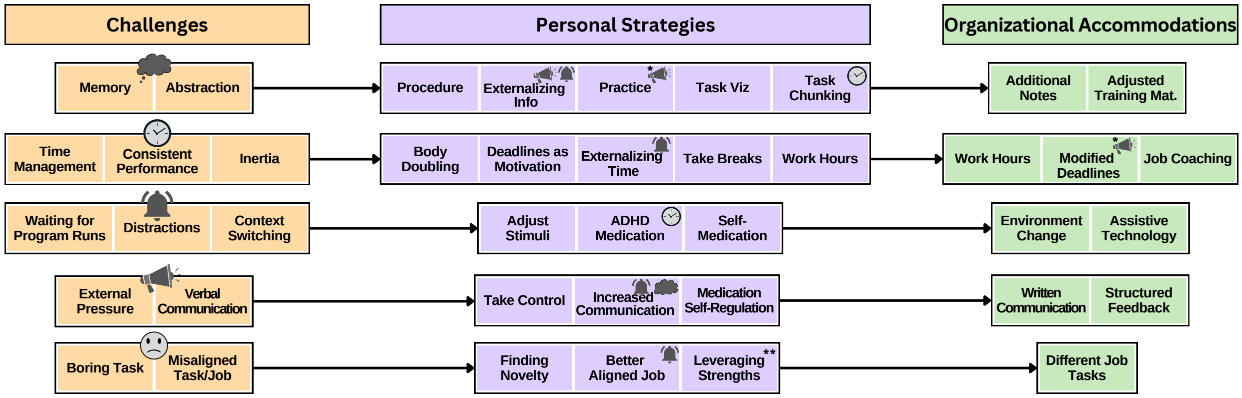

Fig. 1. Our axial coding of challenges, personal strategies, and organizational accommodations relevant to ADHD programmer experiences. The five challenge types are: **Cognition** (Memory, Abstraction), **Time** (Time Management, Consistent Performance, Inertia), **Distractions** (Waiting for Program Runs, Distractions, Context Switching), **Communication** (External Pressure, Verbal Communication), and **Alignment** (Boring Task, Misaligned Job/Task). We connect these five challenge types to personal strategies and organizational accommodations that can be used to mitigate them. Symbols indicate additional challenges for which a strategy or accommodation is also used. **Practice and Modified Deadlines* are connected to the *Communication* challenge in the context of the interview process. ***Leveraging Strengths* refers to seeking jobs, company environments, and working cultures that align with and incorporate the strengths of ADHD programmers (see §V-E).

A. Challenges Involving Cognition: Memory and Abstraction

Building software is memory-intensive; clear mental models are essential for designing software systems [88]. In our qualitative results, ADHD programmers reported struggles with short- or long-term memory (29 posts) and task abstraction (34 posts), e.g., not seeing the big picture beyond the details, or being overwhelmed by of the magnitude of a system.

Impacted Software Tasks. In our survey, ADHD developers reported *challenges with memory* more frequently than NT developers during meetings, design, code review, reading documentation, and brainstorming (see Table III). These validate our qualitative results; for instance, *"daily standups can be hell because I have no idea what I was doing since the last standup"* (D59, or Document ID 59—see Table I). Similarly, ADHD developers perceive *abstraction* more challenging during design, debugging, code review, documentation, and meetings. For example, one commenter struggled with *"breaking down the logic of each ticket without being paralyzed by branches on branches of tangential thoughts"* (D91).

Strategies. In our qualitative analysis, we identified 5 primary strategies that ADHD developers use to support memory and abstraction (see Fig. 1): procedures, externalizing information, practice, task visualization, and task chunking. In our survey, The most commonly-used strategies by ADHD programmers were task chunking (72%), taking meeting notes (40%), task visualization (38%), and procedures (23%) (see Table IV). We delve into our qualitative results below.

ADHD developers *chunk tasks* to keep track of which level of abstraction to focus on. They also *externalized information* by recording meetings, taking notes, and writing code comments. This helped programmers recall context, prepare for interviews, or remember meeting content. *"Are you interviewing on Zoom? Make a…poster of all the things you might need to talk about during the interview and put it behind your webcam on the wall"* (D83). *Practicing* LeetCode was used to memorize content for technical interviews.

Task visualization was helpful for abstraction, especially when translating from design to implementation. *"Talking out my ideas and showing process through pictures…help[s] me collect my thoughts…[and] translate it into syntax"* (D47).

Task ordering *procedures* also helped memory and abstraction. These ranged from to-do lists to workflow-integrated programs, such as issue trackers, GNU Emacs "org mode", or notes apps. *"[I] have a mental list of 'common pitfalls'…I read over the changes once looking only for potential concurrency problems; then paying attention to names; [etc.]"* (D92).

Accommodations. In our qualitative analysis, we identified 2 accommodations provided to ADHD developers to help with memory and abstraction-heavy tasks: *additional notes* and *adjusted training*. Some developers desired support in *acquiring meeting notes* including *"written recaps after 1 [on] 1s detailing action items, expectations, and deadlines"* (D96). Accommodations were also sometimes needed for permission to record meetings. Programmers also received *extra or modified training* to help with task abstraction. *"Instead of being given a huge and dry book, they paid for a more interactive online course."* (D5) Both of these accommodations validated in our survey: additional notes (16% desired, 9% granted) and adjusted training materials (6% desired, 4% granted).

> ADHD developers struggle with **memory** and **task abstraction**, especially in *meetings*, *software design*, *code review*, and *technical interviews* (RQ1). Strategies include breaking tasks into smaller pieces, externalizing information, and following procedures (RQ2, RQ3). Accommodations include extra notes or modified training (RQ2, RQ3).

B. Challenges Involving Time: Time Management and Inertia

Effective time management is essential for software engineering [11], [44]. ADHD developers reported struggles with

TABLE III. Work challenges for ADHD programmers from our qualitative data (left three columns), triangulated with our survey (right two columns). *Associated SE Tasks* are software tasks that co-occur with a challenge in our qualitative data, ordered by prevalence. **Bolded** tasks are validated by our survey. *Significant Tasks for ADHD Developers* were more challenging for ADHD developers than NT developers in our survey (BH-adjusted $p < .05$ when holding other variables constant, see §III-B). For each, we report an *Odds Ratio* (OR). This is the odds of an ADHD participant rating themselves more highly on a question's scale as compared to a NT participant. For example, the odds of an ADHD developer struggling *more frequently* with time management during software design is 4.42 times that of a NT developer. Some qualitative tasks were not in our survey, and are thus not possible to validate (see §III-B). *Survey Questions* are modified for space.

Challenge	Indicative Quote	Associated SE Tasks	Survey Question	Significant Tasks for ADHD Developers (Task: OR)
Memory	"I am supposed to be able to sit in long meetings, and remember all the information afterwards, despite there being no notes or recordings." (D41)	**Meetings**, Interviews, **Documentation**, **Design**, **Code Review**	How often do you struggle with **memory** during each software task?	Meetings: 2.4, Design: 1.9, Code Review: 1.8, Reading Documentation: 1.8, Brainstorming: 1.8
Abstraction	"I've had many experiences of going down a rabbit hole with a JIRA ticket, spending days to come up with a pull request and then had to throw it all away because I misunderstood the requirements." (D13)	Coding, **Code Review**, Interviews, Version Control, **Documentation**, **Design**, **Meetings**, Refactoring, Requirements Elicitation, Testing	How often do you struggle with getting **lost in the small details** of each software task?	Design: 3.2, Debugging: 2.9, Code Review: 2.9, Reading Documentation: 2.4, Writing Documentation: 2.4, Meetings: 2.2
Time Management	"One of my biggest issues is I get sucked in to my projects so much that I lose track of time. It's even worse when I am trying to debug something." (D26)	**Meetings**, Interviews, Agile, **Debugging**	How often do you struggle with **time management** during each software task?	Design: 4.4, Brainstorming: 3.8, Code Review: 3.2, Writing Documentation: 3.1, Debugging: 3.1, Reading Documentation: 3.1, Meetings: 3.0
Inertia	"I am always present at the meetings but when I have to develop and deliver code, sometimes I beg myself [to sit] in the chair. . . I postpone everything until the deadline." (D85)	Version Control, Coding	How often do you struggle with **starting or continuing** each software task?	Writing Documentation: 3.0, Code Review: 2.4, Reading Documentation: 2.3, Debugging: 1.8
Distractions	"It is such a bad habit of mine to cmd+tab to chrome and open facebook in a new tab while things compile and get sucked in." (D66)	**Meetings**, Interviews, Debugging, **Code Review**	How often do you struggle with **getting distracted** during each software task?	Meetings: 3.7, Writing Documentation: 3.4, Code Review: 3.0, Reading Documentation: 2.8, Design: 2.2, Brainstorming: 1.8
Communication	"My boss called me in today for some feedback. We don't have good communication, and as he pointed out. . . we always have misunderstandings that trigger uncomfortable feelings." (D9)	Meetings, Interviews, Agile	How often do you struggle to **communicate** at your workplace?	Overall: 3.0†
Disliked or Boring Tasks	"At this point I can't even practice leetcode anymore, I look at the problem and my brain goes to sleep like 'yeah, that's boring...' and I literally can't do it. . . " (D43)	Meetings, Interviews, Documentation, Agile, Refactoring, Coding	How motivated are you to complete rote or simple programming tasks?	Overall: .42, adjusted = 2.4*

†Participants in the Other ND group also struggled more frequently with communication ($OR = 2.38$, $p = .024$). No other questions in this table were significant for participants in the Other ND group. *The adjusted (inverse) odds ratio here describes ADHD participants' odds of being *less motivated*.

time management and related challenges in our qualitative analysis: 30 posts mentioned time management explicitly, 26 mentioned inertia (i.e., trouble starting or finishing tasks), and 12 mentioned inconsistent performance.

Impacted Software Tasks. In our survey, ADHD developers struggled more frequently with *time management* during design, brainstorming, code review, documentation writing, debugging, documentation reading, and meetings ($p < 0.05$). In our qualitative analysis, commenters mentioned time management alongside meetings, interviews, Agile, and debugging. *"One of my biggest issues is I get sucked in to my projects so much that I lose track of time. It's even worse when I am trying to debug something. I could easily get stuck on a problem for hours without realizing how much time has passed. . . "* (D26)

In our qualitative analysis, *inertia* was a general problem that cropped up with new projects or tasks (due to overwhelm), or almost-finished ones (due to the end stages of a project or task being boring). In our survey, ADHD developers reported struggling more frequently than NT developers with inertia during writing documentation, code review, reading documentation, and debugging. It may be that writing documentation is especially hard to start because the process of coding was not traceable, and hard to finish because it is subjectively tedious.

With less supportive managers, consistent performance could impact stand-ups with some developers sleeping through meetings or afraid to report during low-performance weeks. However, as some programmers comfortably hit deadlines but had inconsistent performance week-to-week, it could become a non-issue. *"I told my tech lead I slack off a lot. . . He didn't notice because I deliver high quality work at a blistering pace. . . [and] if 'slacking off'. . . helps me in my work and keeps me satisfied, he doesn't mind if I do it"* (D25).

Strategies. We identified 5 primary strategies in our qualitative analysis used by ADHD programmers to help with time-related challenges: body doubling, using deadlines, externalizing time, taking breaks, and adjusting work hours. The most common strategies reported by ADHD developers in our survey were taking breaks (51%), coworking or "body doubling"

(35%), using routines or deadlines (33%), and timers (31%) (see Table IV). We describe our qualitative results below.

Developers often *took breaks* or *adjusted their schedule* to better manage their time. One commenter encouraged another to *"walk away from your desk for a few minutes. If possible, go to an area that is visually/thematically/environmentally different…and clear your head"* (D69).

Several strategies helped increase motivation. For example, *coworking* (sometimes referred to as "body doubling" [24]) was used as motivation to work and increase productivity. *"Something you could try is finding a focus buddy in your team…where you set up games/competitions/accountability"* (D68). *Deadlines* were also used as motivation, gamifying work and making it more exciting. *"If I have a meeting at 10, then I only have 1 hr before that to get a task done. It becomes a challenge to see if I can complete that task"* (D13).

Commenters said that *externalizing time* (e.g., using timers) helped them stay in a flow state. One common method was the "pomodoro" technique, which combines work sprints with breaks [16]. *"25 min work, the timer rings, 5 minutes mandatory break…I made a little bash script that time my cycles and notify me when it's break time or work time"* (D15).

The use of ADHD medication and task chunking was also reported to help with managing time (see §V-A and §V-C).

Accommodations. We qualitatively identified 3 time-related accommodations used by ADHD programmers that we validated in our survey (see Table IV): adjusted schedules (41% desired, 35% granted), relaxed deadlines (26% desired, 19% granted), and help from personnel (e.g., job coaches or therapists, 10% desired, 4% granted).

Some programmers desired or were granted changes in *work hours* or vacation time. For example, one commenter advised another that *"…it's not unreasonable to ask for a flexible working hours as an accommodation"* (D7). Developers also desired or were granted *delayed or incremental deadlines*. As with *task chunking*, incremental deadlines motivate programmers to start or complete work. Since ADHD programmers can *"…struggle with driving progress on long-term projects when they don't have clear milestones,…asking for an accommodation for more structured work might be helpful"* (D96).

A few commenters also mentioned personnel or *job coaching* as a helpful accommodation. *"If your company will pay for job coaching as an accommodation, it's almost always the best way to go because a job coach will partner with you to co-design new tools/strategies for task initiation that are tailored for you"* (D96). Given the diverse work improvements that can be explored via job coaching, this accommodation may generally benefit challenges beyond time management.

> ADHD developers struggle with **time management** and **inertia**, especially during *software design*, *writing documentation*, and *code review* (RQ1). Strategies include working with others, and using deadlines as motivation (RQ2, RQ3). Accommodations include flexible work hours, adjustable deadlines, and job coaching (RQ2, RQ3).

C. Challenges Involving Distractions: Focus and Environment

Sustained focus and flow is essential for software productivity [27], [51]. In our qualitative analysis, we found that ADHD programmers experience distractions (42 posts), trouble context switching (15 posts), and losing focus while waiting for program runs (4 posts). We found three categories of distractions: environmental (e.g., audio or visual noise), virtual (e.g., notifications), and internal (i.e., wandering thoughts).

Impacted Software Tasks. Distractions were mentioned most frequently alongside meetings, as well as interviews, debugging, code review, and design. One commenter mentioned, *…last august I got a job but they said I couldn't pay attention in meetings and got fired."* (D38) Context switching was mentioned most frequently alongside meetings, as well as code review, interviews, and version control. *"It's a huge pet peeve of mine when someone asks me to jump on a quick call…I'll need to schedule the call and have to interrupt what I was doing.. and after the call, I have to get back into my flow state again"* (D81). Waiting for program runs was challenging during testing and version control. *"Compilation. Running Tests. Starting up your dev server. 5 Seconds is effing forever for me. Whatever I was thinking is just gone, or I ALT+TAB and fall down a rabbit hole"* (D66).

Compared to NT developers, ADHD developers in our survey reported more frequent distractions during meetings, documentation, code review, design, and brainstorming ($p < 0.05$). ADHD developers also reported more trouble with *context switching* at work ($OR = 3.1$, $p < .001$).

Strategies. We identified 3 strategies ADHD developers use to mitigate distractions: adjusting stimuli, prescribed medication, and self-medication. In our survey, the most common way ADHD developers adjusted stimuli were adding stimuli (e.g., listening to music or fidgeting [84]) (72%), blocking distractions (67%), and changing their environment (54%). Of respondents with ADHD, 40% used prescribed medication. We did not survey self-medication, as prior work has uncovered barriers to medication for ADHD developers and how they may compensate [63]. We now describe the primary actionable strategies we identified in our qualitative analysis.

Commenters often *adjusted stimuli* to avoid getting distracted. This included adding stimuli, blocking noise, or modifying work environments. *"Having something to keep me busy in a way that doesn't pull my attention from the meeting is a must. A notebook where I can take notes or doodle is great."* (D13) A common method for blocking distractions in our sample was noise-cancelling headphones, especially in the context of open office plans. *"I have noise cancelling headphones and can work with them no matter how loud it is around me."* (D5) To help with waiting for program runs, some commenters built scripts that remind them to return to their work when it finished. *"When I need to run a CLI command that takes a while, I just add '&& say 'All done, check your terminal" at the end, and the computer literally tells me to go check"* (D66) (see externalizing time, §V-B).

Commenters also externalized information to remain on task, find a job that did not have as many distractors, and increase communication about their need to not be disturbed (see §V-A, §V-E, and §V-D for details on these strategies).

Accommodations. We identified 2 primary accommodations that ADHD developers desire and are granted to help with distractions. In our survey, the most commonly desired and granted accommodations for ADHD developers for these challenges were adjusted working spaces (18% desired, 13% granted) and assistive technology (11% desired, 7% granted).

In our qualitative analysis, commenters desired or were granted *adjusted environments* in the form of quiet offices, blocked distractions (e.g., pull request notifications), or work location. Some commenters preferred working from home to block distractions, while others preferred working in person to be motivated by others around them (i.e., body doubling). Commenters also mentioned *assistive technology*, such as headphones, sit-stand desks, or captions to mitigate distractions. For example, one commenter's company offers *"a larger monitor to help maintain focus"* (D1).

> Programmers with ADHD struggle with **distractions**, especially during *meetings*, writing *documentation*, or *waiting for software execution* (RQ1). Developers added stimuli (e.g., fidgeted), blocked stimuli (e.g., with headphones), and used medication (RQ2, RQ3). Helpful accommodations included adjusted working environments and assistive technology (RQ2, RQ3).

D. Challenges Involving Communication: Misunderstandings

Communication within and between software teams is essential for effective development [21]. We found that ADHD programmers struggle with misaligned communication (22 posts) and pressure from others (10 posts). In our survey, both ADHD and Other ND programmers struggled with communication more than did NT programmers ($p < 0.05$, Table III).

Impacted Software Tasks. In our qualitative analysis, verbal misunderstandings were reported during meetings, interviews, and collaborative tasks (e.g., code review or mob programming). *"The tasks were assigned verbally, which is my weakest form of communication…I spent months constructing an elaborate test suite for one project after I thought I was given the request to do so, only to be told that it was entirely unnecessary"* (D34). Commenters also reported that external pressure (e.g., timed tasks or confusing demands) hurt performance during interviews and code review: *"on the spot with someone I don't know, I'm not going to do as well"* (D47).

Strategies. In our qualitative analysis, we identified 3 primary strategies used by ADHD programmers to mitigate for challenges with communication: taking control, increasing or adjusting communication, and self-regulating medication use.

ADHD developers *increased communication* to clear up misunderstandings and communicate needs. *"I have a sign with 'Sorry, I'm busy' that I put on my desk.… When I'm focused and people want to talk to me, I try to explain they should only drag me from my hyperfocus when it's urgent."* (D24). In our survey, 51% of ADHD participants used frequent communication to aid with misunderstandings.

Commenters also *took control* in meetings to clear up misunderstandings or make them run efficiently. *"If I'm feeling lost,…, [I] interrupt and ask a lot of questions…They appreciate being heard, and neither of us suffer the consequences of miscommunication, so I'm not shy about it"* (D23).

Accommodations. We identified 2 communication-related accommodations used by ADHD developers. Commenters from our qualitative data desired or received accommodations for *written communication* and *structured feedback* from bosses as it was easier to interpret and act on. Written communication, such as *"using emails or instant messages instead of meetings wherever possible"* (D68), removed the need to parse and recall speech. The use of a different mode of communication generalized in our survey (29% desired, 17% granted).

> ADHD programmers struggle with **communication** in *meetings* and *interviews*, often due to misunderstandings during verbal conversations (RQ1). Strategies include explicitly communicating needs or using written communication (RQ2, RQ3), which were supported by the accommodations of an adjusted communication mode or structured feedback (RQ2, RQ3).

E. Challenges Involving Alignment: Misaligned Tasks/Jobs

We refer to activities with social, environmental, or physical barriers for ADHD developers as *misaligned*. 28 posts mentioned misaligned tasks and 7 mentioned misaligned jobs.

Impacted Software Tasks. In our qualitative analysis, feelings of misalignment were expressed during timed technical interviews, daily standup, and refactoring with commenters finding them stressful, repetitive, or inefficient. Misaligned jobs had rigid schedules, required intense attention to details, or heavily surveilled employees. *"For me, everyone leaves a bit after my mid day, with on and off meetings between 8-1ish…I am at peak dysfunction around that time, always have been"* (D13).

We also found that *boredom* was common during misaligned tasks, especially during meetings or documentation writing. Boredom can lead to lower productivity [18], and ADHD *"makes a challenge for things like documentation or highly repetitive/low effort tasks"* (D34). This validated in our survey: ADHD developers were less motivated to complete rote or simple tasks than NT developers ($p < 0.05$).

Strategies. We identified 2 strategies that ADHD developers use to help with alignment: finding novelty and obtaining a better fitting job. One of these validated in our survey: 60% of ADHD participants have chosen a job based on work-style fit (vs. 46% of NT, $p < 0.05$, see Table IV). We also identified strengths that ADHD developers can better leverage when in an aligned environment including creativity, and hyperfocus.

In our qualitative analysis, commenters sought *novelty* to mitigate boredom. *"Alternating WFH and office is a spectacular way to keep your brain thinking 'Oh yay I'm moving!'"* (D46). Commenters also *sought better aligned jobs*, often leaving current jobs to do so. Such jobs had flexibility in task allocation and scheduling, work-from-home support, limited distractions, and general good engineering practices.

Leveraging Strengths. We identified 6 strengths of ADHD developers that are supported by aligned environments: creativity, working with novelty, crisis management, breadth of knowledge, hyperfocus, and better aligned tasks.

Commenters reported *creativity* during brainstorming and problem solving. Surveyed ADHD developers reported more creativity during programming tasks (vs. NT, $OR = 2.7$, $p < .001$). Commenters also thrived with *novelty*, noting it can give *"…those with ADHD a large competitive advantage"* (D21). *Crisis management* was also a strength. *"I actually get better as the situation gets worse. It's the reason why I always get promoted to manager/director"* (D12). Commenters also reported *breadth of knowledge* as a strength. *"[As] a jack of all trades[,…I] help [my team] understand why and how their task fits into the big picture"* (D2).

At aligned jobs, commenters reported the ability to hyperfocus on a specific task, increasing productivity. *"I love to CNC program…Where it will take someone a day to release 1 thing, I usually have 2-2.5 done…I hyperfocus on it and get what needs to be done, and I have a lot of fun"* (D76). ADHD developers in our survey reported hyperfocusing more during software tasks than NT programmers ($OR = 1.9$, $p = .002$).

In terms of aligned tasks, ADHD developers reported more skill at brainstorming (vs. NT, $OR = 1.6$, $p = 0.02$). Both ADHD ($OR = 1.6$, $p = 0.03$) and Other ND ($OR = 2.0$, $p = 0.03$) reported more skill at debugging. We note that participants that *only* had ADHD *did not* report higher debugging ability, associating this result with autistic participants. These were expected results; the creativity of people with ADHD [9] may lead to skill with brainstorming, and attention to detail and systematizing have been linked to autism [2]. Overall, adapting software workplaces and assigned tasks to highlight strengths and remove blockers for ADHD developers may be beneficial for both developers and their workplaces.

Accommodations. Our survey validated one accommodation used for challenges involving alignment: *different task assignment* (desired: 31%, granted: 22%). In our qualitative analysis, some commenters negotiated with their manager for more challenging work. Many commenters also desired modified technical interviews. While some worried that asking would impact their hiring chances, others said *"yes, you can and should ask…I had people at my company ask for a different style of assessment and it's fine"* (D43).

> ADHD programmers encounter jobs that are **misaligned** with their working styles (e.g., emphasize simple tasks, have rigid schedules) (RQ1). This can cause developers to switch jobs or desire different tasks, including for *interviews* (RQ2, RQ3).

VI. Results: ADHD vs. Non-ADHD Developers

Though the frequency or qualia of challenges experienced by ADHD developers may differ for non-ADHD developers, our survey may have generalizable insights. The "curb-cut effect"—when interventions for one disadvantaged group benefit a broader population [8]—occurs in many contexts [25], [41], [73]. We focus on how identified challenges affect non-ADHD programmers and strategies and accommodations could be introduced for more widespread support (RQ4).

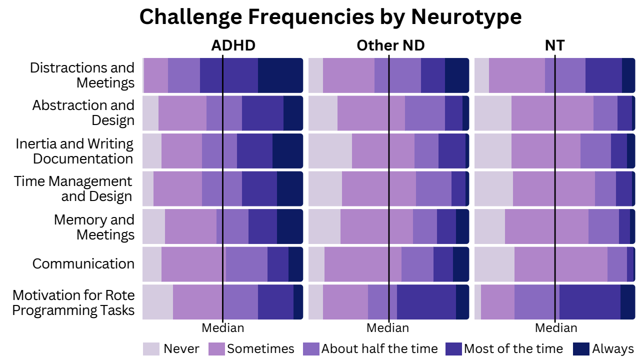

Fig. 2. Frequencies for software/task pairs from each challenge type that were the most challenging for ADHD programmers (vs. NT, $p < 0.05$, $OR > 2$). Communication by software task was not asked about in our survey, and thus stands alone.

Although all identified challenge types were experienced more frequently by ADHD developers ($p < 0.05$), they were also relevant for NT and Other ND. Fig. 2 shows the frequencies by neurotype for the challenge/task pairs that impacted ADHD developers the most disproportionately. Notably, the median frequency for NT and Other ND was consistently "Sometimes" or "About half the time", indicating that non-ADHD developers experience these challenges a non-trivial amount of the time. Therefore, providing organizational support could be beneficial for more than just ADHD developers.

Table IV shows strategy frequency by neurotype. Some techniques such as blocking distractions, coworking, or medication were used more frequently by ADHD developers ($p < 0.05$). However, many strategies were common regardless of neurotype, such as task chunking (80% of NT) or taking breaks (53% of NT). Even if more common for ADHD developers, NT developers were also likely to block distractions (49%) or add stimuli (e.g., fidgeted, 41%). This suggests supporting these strategies may allow softwar companies to leverage a potential curb-cut effect, leading to broader productivity gains.

We have preliminary evidence that organizational support could also generalize: 58% of NT participants desired at least one organizational accommodation. While some accommodations were desired more by ADHD programmers (e.g., different job task or communication mode, $p < 0.05$), others were desired by developers at all neurotypes at similar rates. For example, 35% of NT participants desired an adjusted schedule (compared to 41% of ADHD participants, $p = .41$, see Table IV). Implementing such accommodations as publicly-accessible support options may thus benefit developers regardless of neurotype. We discuss potential next steps in §VIII.

TABLE IV. Percent of ADHD, Other ND (O. ND), and NT survey participants that use each strategy or desire each accommodation, ordered by prevalence across all responses. Items used or desired significantly more by a group are bolded and marked teal, while those used or desired less are tan (χ^2-test, BH-corrected $p < 0.05$).

Strategy	Overall	ADHD	O. ND	NT
Task chunking	75%	72%	68%	80%
Blocking distractions	59%	**67%**	57%	49%
Adding stimuli	58%	**72%**	58%	41%
Choosing a better fitting job	54%	**60%**	55%	46%
Taking breaks	53%	51%	60%	53%
Changing environment	53%	54%	55%	49%
Frequent communication	51%	51%	48%	52%
Recording/taking meeting notes	42%	40%	52%	42%
Task visualization	39%	38%	45%	37%
Regular routines/deadlines	32%	33%	28%	31%
Timers (e.g., pomodoro)	28%	31%	27%	24%
Coworking or "body doubling"	25%	**35%**	12%	18%
Prescribed medication	24%	**40%**	12%	7%
Outlined procedures	21%	23%	22%	18%
Desired Accommodation†				
None	40%	38%	39%	42%
Adjusted schedule	39%	41%	44%	35%
Different job tasks	25%	**31%**	25%	17%
Relaxed deadlines	23%	26%	19%	20%
Different communication mode	22%	**29%**	26%	12%
Adjusted working space	19%	18%	19%	21%
Additional notes or materials	13%	16%	11%	11%
Assistive technology	9%	11%	7%	7%
Help from personnel	8%	10%	9%	4%
Extra or modified training	5%	6%	7%	4%

†In §III, we inline the percent of accommodations granted for ADHD developers. All were granted at a lower rate than desired.

VII. Threats to Confirmability and Validity

Qualitative Threats. Our qualitative findings may not be *credible* to ADHD programmers or other researchers [32]. To mitigate this, we triangulate with prior work, between authors, and with our survey. In addition, before our analysis we read hundreds of posts on `r/ADHD_Programmers`, familiarizing ourselves with the community via *prolonged engagement* [32].

Our qualitative results may not *transfer* to other contexts due to negativity bias in online communities [89], unique Reddit aspects (e.g., region skew), and low self-efficacy seen among ADHD populations [62]. We mitigate this by validating our results through a large-scale survey of professional programmers where we ask about strengths and measure wellbeing, which can inform how people cope with disability [33].

Finally, qualitative results must have a traceable process [32]). We collected meeting notes, records of analysis passes, and codebook versions. We do not publicly release our data for ethical reasons, but we include all processes-related documents (including codebooks) in our replication package.

Quantitative Threats. We use our survey to validate our qualitative findings in a different context. However, our survey may not *generalize* due to self-selection bias and data quality. For example, participants with non-standard ADHD-related experiences might be more likely to participate. We note that key sample demographics align with those of GitHub [90], partially mitigating this concern. To ensure data quality, we use previously-validated measures when possible, and also employ consistency and time-related data quality checks.

Our participant groupings may also impact our statistical results. Due to the intersectional nature of our qualitative data, we combine diagnosed and self-diagnosed ADHD programmers and do not consider multiple diagnoses. We gain confidence in our approach by conducting several additional analyses with different groupings, where we find minimal impacts on our overall conclusions. For the interested reader, we include these additional analyses in our replication package.

VIII. Discussion

We contextualize our results with regard to related work on neurodivergent developers and general challenges faced by software developers. We then discuss broad categories of interventions grounded in our findings and future work.

A. Comparison to Related Work

We consider related work at the intersection of ADHD and professional programming. Morris *et al.* identified challenges disproportionately impacting neurodivergent programmers via a survey of 846 software engineers (59 neurodivergent) [59]. They found that 13.6% of neurodivergent participants desired, but had not asked for or received accommodations. While 38 participants had ADHD, they grouped all neurodivergent participants in their analysis. Some identified challenges were similar to ours, including difficulties focusing on tasks and with workplace communication. Common strengths found between our work and theirs were hyperfocus and creativity ("divergent thinking"). However, many identified themes were different (e.g., challenges with "handling changes in routine" and a strength in "detecting patterns in code") [59].

In a study on substance use in software, Newman *et al.* found that 15 out of 26 participants had ADHD and used medication (e.g., Adderall) as a strategy at work, reporting improved productivity at most software tasks [63]. Kasatskii *et al.* found a significant effect of visual noise and ADHD symptoms on debugging efficiency, but the effect was not direct [42]. In a qualitative study with 5 participants, Gama *et al.* found that Agile practices such as effort estimates may reduce efficiency for ADHD developers [31].

Liebel *et al.* [49] studied the challenges, strategies, and strengths of ADHD developers in a case study of 19 interviewees. Several of the challenges and strategies we identified replicate those in the case study, including distractions, time management, medication, and taking breaks. We also identified new challenges (e.g., abstraction and waiting for program runs) and strategies (e.g., task chunking and using deadlines as motivation). We are also the first to identify software task-specific challenges for ADHD programmers, to research accommodations for ADHD programmers, to map strategies and accommodations to challenges they may mitigate for, and to generalize these findings via a large-scale survey.

B. Contextualization with General Software Challenges

For all challenge types, we find evidence from our survey that ADHD developers are affected more frequently than neurotypical developers. These broad challenge types may affect developers generally, meaning that interventions which support ADHD developers could lead to widespread benefit [8].

Memory is vital for navigating large codebases. Limitations of human memory can impact software engineering and development tools can be used as aids [66]. Working in a distracting environment can also challenge neurotypical programmers. In a control trial, Ma *et al.* found that both automated and interpersonal distractions can lead to increased physiological stress during software development [51]. Developers can also struggle with time management. One study on time pressure in software engineering found that time pressure, often caused by errors in cost estimation, can lead to decreased software quality [44]. Poor communication within software teams [21] or unaligned task allocation [52] can decrease team productivity.

Our survey demonstrates that these challenges impact ADHD programmers disproportionately. We also found that they can impact ADHD programmers in unique ways. For example, while communication can be a challenge for software teams in general, the preference for written communication is more associated with ADHD; general works focus on the amount of communication rather than the modality [21]. Similarly, a distraction for someone with ADHD, such as a passing thought, might not be a distraction for someone neurotypical (prior work focuses on environmental distractions [51]).

C. Interventions and Future Work

Given ADHD developers' strengths and challenges, we provide recommendations for interventions that may help them excel. Since "official" accommodations can be risky to obtain and are underused [4], [22], [59], we propose general interventions that have the potential to not only help ADHD software engineers, but also software engineers at large. We propose *changes to workplace structures*, such as incremental deadlines and information scaffolding; *flexibility* in task allocation, work hours, and work-from-home status; *time-blocking and asynchronous communication* for meetings; and publicly-available *assistive technology*, such as headphones or "stim objects" (i.e., fidget toys). Many of these align with software engineering best practices. For example, asynchronous communication is important for global software teams [87] and incremental deadlines (often used in Agile) aid with cost estimations and scope creep [70]. We recognize that certain environments may not admit all changes (e.g., a startup may not be able to flexibly allocate tasks). However, we believe that at least one could be implemented in most environments.

Based off our findings, we propose technological interventions targeting increasing "flow" (i.e., periods of focused work), which is identifiable via log data [12]. Tools which notify programmers upon program run completion could mitigate for unnecessary context switching; some forum commenters built their own. In-IDE games could also help with wait times, as gamification and fidgeting have been shown to improve focus and learning for people with ADHD [71], [84]. Finally, AI-summarization in meetings (e.g., in [69]) could be used to assist working memory. Distractions during meetings was the most frequent challenge for both ADHD and non-ADHD developers, making it salient to target for widespread change.

Future work may benefit from objective measures of behavioral outcomes with our identified support mechanisms. In this work, we use subjective measures and experiences of ADHD developers; observational or experimental work is needed to see if identified support translates to outcomes (e.g., improvements in accuracy or time efficiency). For tool- or policy-building, we recommend using participatory design [85] to determine what tools ADHD developers find useful.

IX. Conclusion

ADHD is common among software developers. However, ADHD developers can struggle disproportionately with certain software tasks, and support options are not well understood. We conducted a mixed methods study of ADHD professional programmers: a qualitative analysis of 1,658 Reddit posts and comments, and a survey of 493 professional programmers. We propose the first software task-specific mapping between ADHD developers' challenges, strategies, and accommodations. We are also the first to study ADHD developer accommodations, finding that many could support both ADHD and neurotypical programmers alike. Our findings have implications for tools and policies supporting ADHD programmers, along with potential lightweight changes for software workplaces. These include **changes to workplace structures** (e.g., incremental deadlines), **work environment flexibility** (e.g., work from home support), **time-blocked meetings**, **publicly-accessible assistive technology** (e.g., noise-canceling headphones), **task allocation flexibility** (e.g., allowing developers to choose tasks), and **prioritized written communication**. We believe that software workplaces that invest in this way could see increased productivity in not just ADHD developers, but more generally. We hope our work leads to more inclusive software environments for developers of all neurotypes.

Acknowledgments

The authors would like to thank Andrew Grogan-Kaylor, Jodi Forlizzi, Jim Herbsleb, and Bogdan Vasilescu for advice on methods and mentorship. The authors would also like to acknowledge the friendship of their kitty companions, namely: Holly, Pepper, Cayenne, Sage, and Cleo.

This material is based upon work supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE2140739. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

[1] Marios Adamou, Muhammad Arif, Philip Asherson, Tar-Ching Aw, Blanca Bolea, David Coghill, Gísli Guðjónsson, Anne Halmøy, Paul Hodgkins, Ulrich Müller, et al. Occupational issues of adults with ADHD. *BMC psychiatry*, 13:1–7, 2013.

[2] Hala Annabi, Karthika Sundaresan, and Annuska Zolyomi. It's not just about attention to details: Redefining the talents autistic software developers bring to software development. In *50th Hawaii International Conference on System Sciences, HICSS, January 4-7, 2017*, pages 1–10. ScholarSpace / AIS Electronic Library (AISeL), 2017.

[3] Philip Asherson, Ron Akehurst, JJ Sandra Kooij, Michael Huss, Kathleen Beusterien, Rahul Sasané, Shadi Gholizadeh, and Paul Hodgkins. Under diagnosis of adult ADHD: cultural influences and societal burden. *Journal of Attention Disorders*, 16(5_suppl):20S–38S, 2012.

[4] David C Baldridge and Michele L Swift. Withholding requests for disability accommodation: The role of individual differences and disability attributes. *Journal of Management*, 39(3):743–762, 2013.

[5] R. A. Barkley, K. R. Murphy, and M. Fischer. *ADHD in adults: What the science says*. Guilford Press, 2010.

[6] Jason Baumgartner, Savvas Zannettou, Brian Keegan, Megan Squire, and Jeremy Blackburn. The pushshift reddit dataset. In Munmun De Choudhury, Rumi Chunara, Aron Culotta, and Brooke Foucault Welles, editors, *International Conference on Web and Social Media, ICWSM, June 8-11, 2020*, pages 830–839. AAAI Press, 2020.

[7] Craig M Bennett, Michael B Miller, and George L Wolford. Neural correlates of interspecies perspective taking in the post-mortem atlantic salmon: An argument for multiple comparisons correction. *Neuroimage*, 47(Suppl 1):S125, 2009.

[8] Angela Glover Blackwell. The curb-cut effect. *Stanford Social Innovation Review*, 15(1):28–33, 2016.

[9] Nathalie Boot, Barbara Nevicka, and Matthijs Baas. Creativity in ADHD: goal-directed motivation and domain specificity. *Journal of attention disorders*, 24(13):1857–1866, 2020.

[10] Rollin Brant. Assessing proportionality in the proportional odds model for ordinal logistic regression. *Biometrics*, 46(4):1171–1178, 1990.

[11] Frederick C Brooks. *The Mythical Man-Month (1975)*. Addison-Wesley, 2021.

[12] Adam Brown, Sarah D'Angelo, Ben Holtz, Ciera Jaspan, and Collin Green. Using logs data to identify when software engineers experience flow or focused work. In *CHI Conference on Human Factors in Computing Systems*, pages 1–12, 2023.

[13] Margaret M. Burnett, Anicia Peters, Charles Hill, and Noha Elarief. Finding gender-inclusiveness software issues with GenderMag: A field investigation. In Jofish Kaye, Allison Druin, Cliff Lampe, Dan Morris, and Juan Pablo Hourcade, editors, *Conference on Human Factors in Computing Systems (CHI), May 7-12, 2016*, pages 2586–2598. ACM, 2016.

[14] John L Campbell, Charles Quincy, Jordan Osserman, and Ove K Pedersen. Coding in-depth semistructured interviews: Problems of unitization and intercoder reliability and agreement. *Sociological methods & research*, 42(3):294–320, 2013.

[15] Carlos Canela, Anna Buadze, Anish Dube, Dominique Eich, and Michael Liebrenz. Skills and compensation strategies in adult ADHD–a qualitative study. *PloS one*, 12(9):e0184964, 2017.

[16] Francesco Cirillo. *The Pomodoro technique: The acclaimed time-management system that has transformed how we work*. Currency, 2018.

[17] Jacob Cohen. *Statistical power analysis for the behavioral sciences*. routledge, 2013.

[18] Mary L Cummings, Fei Gao, and Kris M Thornburg. Boredom in the workplace: A new look at an old problem. *Human factors*, 58(2):279–300, 2016.

[19] Anastasia Danilova, Alena Naiakshina, Stefan Horstmann, and Matthew Smith. Do you really code? designing and evaluating screening questions for online surveys with programmers. In *43rd IEEE/ACM International Conference on Software Engineering, ICSE 2021, Madrid, Spain, 22-30 May 2021*, pages 537–548. IEEE, 2021.

[20] Maitraye Das, John Tang, Kathryn E. Ringland, and Anne Marie Piper. Towards accessible remote work: Understanding work-from-home practices of neurodivergent professionals. *ACM Hum. Comput. Interact.*, 5(CSCW1):183:1–183:30, 2021.

[21] Joanna F Defranco and Philip A Laplante. Review and analysis of software development team communication research. *IEEE Transactions on Professional Communication*, 60(2):165–182, 2017.

[22] Eros R DeSouza, Eric D Wesselmann, and Dan Ispas. Workplace discrimination against sexual minorities: Subtle and not-so-subtle. *Canadian Journal Of Administrative Sciences/Revue canadienne des sciences de l'administration*, 34(2):121–132, 2017.

[23] Nicole M Deterding and Mary C Waters. Flexible coding of in-depth interviews: A twenty-first-century approach. *Sociological methods & research*, 50(2):708–739, 2021.

[24] Tessa Eagle, Leya Breanna Baltaxe-Admony, and Kathryn E Ringland. Proposing body doubling as a continuum of space/time and mutuality: An investigation with neurodivergent participants. In *ACM SIGACCESS Conference on Computers and Accessibility*, pages 1–4, 2023.

[25] Elizabeth F Emens. Integrating accommodation. *U. Pa. L. Rev.*, 156:839, 2007.

[26] Madeline Endres, Kevin Boehnke, and Westley Weimer. Hashing it out: A survey of programmers' cannabis usage, perception, and motivation. In *International Conference on Software Engineering*, pages 1107–1119, 2022.

[27] Nicole Forsgren, Margaret-Anne Storey, Chandra Maddila, Thomas Zimmermann, Brian Houck, and Jenna Butler. The space of developer productivity: There's more to it than you think. *Queue*, 19(1):20–48, mar 2021.

[28] Sophie Forster, David J Robertson, Alistair Jennings, Philip Asherson, and Nilli Lavie. Plugging the attention deficit: perceptual load counters increased distraction in ADHD. *Neuropsychology*, 28(1):91, 2014.

[29] Anselm Fuermaier, Lara Tucha, Marah Butzbach, Matthias Weisbrod, Steffen Aschenbrenner, and Oliver Tucha. ADHD at the workplace: ADHD symptoms, diagnostic status, and work-related functioning. *Journal of Neural Transmission*, 128(7):1021–1031, 2021.

[30] Patricia I Fusch Ph D and Lawrence R Ness. Are we there yet? data saturation in qualitative research. *The Qualitative Report*, 20(9):1408–1416, 2015.

[31] Kiev Gama and Aline Lacerda. Understanding and supporting neurodiverse software developers in Agile teams. In *Brazilian Symposium on Software Engineering*, pages 497–502, 2023.

[32] Egon G Guba and Yvonna S Lincoln. *Fourth generation evaluation*. Sage, 1989.

[33] Martin S Hagger and Sheina Orbell. A meta-analytic review of the common-sense model of illness representations. *Psychology and health*, 18(2):141–184, 2003.

[34] Alexander Halavais. Overcoming terms of service: A proposal for ethical distributed research. In *Disinformation and Data Lockdown on Social Platforms*, pages 45–59. Routledge, 2021.

[35] Spencer E Harpe. How to analyze likert and other rating scale data. *Currents in pharmacy teaching and learning*, 7(6):836–850, 2015.

[36] Judith R Harrison, Nora Bunford, Steven W Evans, and Julie Sarno Owens. Educational accommodations for students with behavioral challenges: A systematic review of the literature. *Review of educational research*, 83(4):551–597, 2013.

[37] Andrew F Hayes and Klaus Krippendorff. Answering the call for a standard reliability measure for coding data. *Communication methods and measures*, 1(1):77–89, 2007.

[38] Yu Huang, Denae Ford, and Thomas Zimmermann. Leaving my fingerprints: Motivations and challenges of contributing to OSS for social good. In *2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)*, pages 1020–1032, 2021.

[39] Kathleen E Hupfeld, Tessa R Abagis, and Priti Shah. Living "in the zone": hyperfocus in adult ADHD. *ADHD Attention Deficit and Hyperactivity Disorders*, 11:191–208, 2019.

[40] Glenn D Israel. *Sampling the evidence of extension program impact*. Citeseer, 1992.

[41] Donna M Johnson and Judith A Fox. Creating curb cuts in the classroom: Adapting universal design principles to education. *Curriculum transformation and disability: Implementing universal design in higher education*, pages 7–21, 2003.

[42] Vseslav Kasatskii, Agnia Sergeyuk, Anastasiia Serova, Sergey Titov, and Timofey Bryksin. The effect of perceptual load on performance within IDE in people with ADHD symptoms. In *International Conference on Human-Computer Interaction*, pages 122–141. Springer, 2023.

[43] Ronald C Kessler, Lenard Adler, Minnie Ames, Russell A Barkley, Howard Birnbaum, Paul Greenberg, Joseph A Johnston, Thomas Spencer, and T Bedirhan Üstün. The prevalence and effects of adult attention deficit/hyperactivity disorder on work performance in a nationally representative sample of workers. *Journal of occupational and environmental medicine*, 47(6):565–572, 2005.

[44] Miikka Kuutila, Mika Mäntylä, Umar Farooq, and Maelick Claes. Time pressure in software engineering: A systematic review. *Information and Software Technology*, 121:106257, 2020.

[45] Kate Kysow, Joanne Park, and Charlotte Johnston. The use of compensatory strategies in adults with ADHD symptoms. *ADHD Attention Deficit and Hyperactivity Disorders*, 9:73–88, 2017.

[46] Thomas D LaToza and Brad A Myers. On the importance of understanding the strategies that developers use. In *ICSE Workshop on Cooperative and Human Aspects of Software Engineering*, pages 72–75, 2010.

[47] Jenny T. Liang, Chenyang Yang, and Brad A. Myers. A large-scale survey on the usability of AI programming assistants: Successes and challenges. In *International Conference on Software Engineering*, ICSE '24, New York, NY, USA, 2024. Association for Computing Machinery.

[48] Jenny T. Liang, Chenyang Yang, and Brad A. Myers. A large-scale survey on the usability of AI programming assistants: Successes and challenges. In *International Conference on Software Engineering, ICSE 2024*, pages 52:1–52:13. ACM, 2024.

[49] Grischa Liebel, Noah Langlois, and Kiev Gama. Challenges, strengths, and strategies of software engineers with ADHD: A case study. In *ICSE Software Engineering in Society*, 2024.

[50] Justina Lukat, Jürgen Margraf, Rainer Lutz, William M van der Veld, and Eni S Becker. Psychometric properties of the positive mental health scale (PMH-scale). *BMC psychology*, 4:1–14, 2016.

[51] Yimeng Ma, Yu Huang, and Kevin Leach. Breaking the flow: A study of interruptions during software engineering activities. In *International Conference on Software Engineering*, pages 1–12, 2024.

[52] Zainab Masood, Rashina Hoda, Kelly Blincoe, and Daniela Damian. Like, dislike, or just do it? how developers approach software development tasks. *Information and Software Technology*, 150:106963, 2022.

[53] Ian McChesney and Raymond Bond. Eye tracking analysis of computer program comprehension in programmers with dyslexia. *Empirical Software Engineering*, 24:1109–1154, 2019.

[54] Peter McCullagh. Regression models for ordinal data. *Journal of the Royal Statistical Society: Series B*, 42(2):109–127, 1980.

[55] Nora McDonald, Sarita Schoenebeck, and Andrea Forte. Reliability and inter-rater reliability in qualitative research: Norms and guidelines for CSCW and HCI practice. *ACM Hum.-Comp. Inter.*, CSCW, nov 2019.

[56] Mary L McHugh. The chi-square test of independence. *Biochemia medica*, 23(2):143–149, 2013.

[57] Courtney Miller, Christian Kästner, and Bogdan Vasilescu. "we feel like we're winging it: " A study on navigating open-source dependency abandonment. In Satish Chandra, Kelly Blincoe, and Paolo Tonella, editors, *Foundations of Software Engineering, ESEC/FSE 2023*, pages 1281–1293. ACM, 2023.

[58] Curt B Moore, Nancy H McIntyre, and Stephen E Lanivich. ADHD-related neurodiversity and the entrepreneurial mindset. *Entrepreneurship Theory and Practice*, 45(1):64–91, 2021.

[59] Meredith Ringel Morris, Andrew Begel, and Ben Wiedermann. Understanding the challenges faced by neurodiverse software engineering employees: Towards a more inclusive and productive technical workforce. In *ACM SIGACCESS Conference on computers & accessibility*, pages 173–184, 2015.

[60] Kevin Murphy and Russell A Barkley. Attention deficit hyperactivity disorder adults: comorbidities and adaptive impairments. *Comprehensive psychiatry*, 37(6):393–401, 1996.

[61] Emerson Murphy-Hill, Alberto Elizondo, Ambar Murillo, Marian Harbach, Bogdan Vasilescu, Delphine Carlson, and Florian Dessloch. Gendermag improves discoverability in the field, especially for women: An multi-year case study of suggest edit, a code review feature. In *International Conference on Software Engineering*, pages 1–12, 2024.

[62] Patricia Elizabeth Newark, Marina Elsässer, and Rolf-Dieter Stieglitz. Self-esteem, self-efficacy, and resources in adults with ADHD. *Journal of Attention Disorders*, 20(3):279–290, 2016.

[63] Kaia Newman, Madeline Endres, Westley Weimer, and Brittany Johnson. From organizations to individuals: Psychoactive substance use by professional programmers. In *2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE)*, pages 665–677. IEEE, 2023.

[64] Oxford English Dictionary. neurodiversity, n., March 2024.

[65] Wendy E Parmet. Discrimination and disability: the challenges of the ada. *Law, Medicine and Healthcare*, 18(4):331–344, 1990.

[66] Chris Parnin and Spencer Rugaber. Programmer information needs after memory failure. In Dirk Beyer, Arie van Deursen, and Michael W. Godfrey, editors, *International Conference on Program Comprehension, ICPC, 2012*, pages 123–132. IEEE Computer Society, 2012.

[67] Eric Patton. When diagnosis does not always mean disability: The challenge of employees with Attention Deficit Hyperactivity Disorder (ADHD). *Journal of Workplace Behavioral Health*, 24(3):326–343, 2009.

[68] Sai Teja Peddinti, Keith W Ross, and Justin Cappos. "On the internet, nobody knows you're a dog": a twitter case study of anonymity in social networks. In *ACM conference on Online Social Networks*, pages 83–94, 2014.

[69] Michelle Peng. The best ai note-taking tools for meetings, Jun 2024.

[70] Kai Petersen and Claes Wohlin. The effect of moving from a plan-driven to an incremental software development approach with Agile practices: An industrial case study. *Empirical Software Engineering*, 15:654–693, 2010.

[71] Arman Syah Putra, Harco Leslie Hendric Spits Warnars, Bahtiar Saleh Abbas, Agung Trisetyarso, Wayan Suparta, and Chu-Ho Kang. Gamification in the e-learning process for children with attention deficit hyperactivity disorder (ADHD). In *Indonesian Association for Pattern Recognition International Conference (INAPR)*, pages 182–185, 2018.

[72] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2023.

[73] Kavita Rao and Adam Tanners. Curb cuts in cyberspace: Universal instructional design for online courses. *Journal of Postsecondary Education and Disability*, 24(3):211–229, 2011.

[74] Brittany Reid, Markus Wagner, Marcelo d'Amorim, and Christoph Treude. Software engineering user study recruitment on prolific: An experience report. *arXiv preprint arXiv:2201.05348*, 2022.

[75] Brian Ripley, Bill Venables, Douglas M Bates, Kurt Hornik, Albrecht Gebhardt, David Firth, and Maintainer Brian Ripley. Package 'mass'. *CRAN R*, 538:112–115, 2013.

[76] Randall Robbins. The untapped potential of the ADHD employee in the workplace. *Cogent Business & Management*, 4(1):1271384, 2017.

[77] Ying Rong, Chang-Jiang Yang, Ye Jin, and Yue Wang. Prevalence of Attention-Deficit/Hyperactivity Disorder in individuals with Autism Spectrum Disorder: A meta-analysis. *Research in Autism Spectrum Disorders*, 83:101759, 2021.

[78] Elias Sarkis. Addressing Attention-Deficit/Hyperactivity Disorder in the workplace. *Postgraduate medicine*, 126(5):25–30, 2014.

[79] Benjamin Schlegel and Marco Steenbergen. Package 'brant'. *CRAN R*, 2020.

[80] Cliff Scott and Melissa Medaugh. Axial coding. *The international encyclopedia of communication research methods*, 10:9781118901731, 2017.

[81] Jane Ann Sedgwick, Andrew Merwood, and Philip Asherson. The positive aspects of Attention Deficit Hyperactivity Disorder: a qualitative investigation of successful adults with ADHD. *ADHD Attention Deficit and Hyperactivity Disorders*, 11(3):241–253, 2019.

[82] Tom Shakespeare et al. The social model of disability. *The disability studies reader*, 2(3):197–204, 2006.

[83] Jennifer M Sims. A brief review of the belmont report. *Dimensions of critical care nursing*, 29(4):173–174, 2010.

[84] Ha Min Son, Catrina Andaya Calub, Boyang Fan, J Faye Dixon, Shahbaz Rezaei, Jared Borden, Julie B Schweitzer, and Xin Liu. A quant. analysis of fidgeting in ADHD and its relation to performance and sustained attention on a cognitive task. *Frontiers in Psychiatry*, 15:1394096, 2024.

[85] Clay Spinuzzi. The methodology of participatory design. *Technical communication*, 52(2):163–174, 2005.

[86] Stack Overflow. Stack overflow developer survey 2022. https://survey.stackoverflow.co/2022/, 2022. Accessed: 2024-07-28.

[87] Viktoria Stray and Nils Brede Moe. Understanding coordination in global software engineering: A mixed-methods study on the use of meetings and slack. *J. Syst. Softw.*, 170:110717, 2020.

[88] Antony Tang, Maryam Razavian, Barbara Paech, and Tom-Michael Hesse. Human aspects in software architecture decision making: a literature review. In *2017 IEEE International Conference on Software Architecture (ICSA)*, pages 107–116. IEEE, 2017.

[89] Sho Tsugawa and Hiroyuki Ohsaki. Negative messages spread rapidly and widely on social media. In *ACM conference on online social networks*, pages 151–160, 2015.

[90] Bogdan Vasilescu, Daryl Posnett, Baishakhi Ray, Mark GJ van den Brand, Alexander Serebrenik, Premkumar Devanbu, and Vladimir Filkov. Gender and tenure diversity in GitHub teams. In *ACM conference on human factors in computing systems*, pages 3789–3798, 2015.

[91] Erik G Willcutt. The prevalence of DSM-IV Attention-Deficit / Hyperactivity Disorder: a meta-analytic review. *Neurotherapeutics*, 9(3):490–499, 2012.