

Tether: A Personalized Support Assistant for Software Engineers with ADHD

Aarsh Shah

University of Calgary
Calgary, AB, Canada
aarsh.shah@ucalgary.ca

Cleyton Magalhaes

UFRPE
Recife, PE, Brazil
cleyton.vanut@ufrpe.br

Kiev Gama

CIn-UFPE
Recife, PE, Brazil
kiev@cin.ufpe.br

Ronnie de Souza Santos

University of Calgary
Calgary, AB, Canada
ronnie.desouzasantos@ucalgary.ca

Abstract—Equity, diversity, and inclusion in software engineering often overlook neurodiversity, particularly the experiences of developers with Attention Deficit Hyperactivity Disorder (ADHD). Despite the growing awareness about that population in SE, few tools are designed to support their cognitive challenges (e.g., sustained attention, task initiation, self-regulation) within development workflows. We present Tether, an LLM-powered desktop application designed to support software engineers with ADHD by delivering adaptive, context-aware assistance. Drawing from engineering research methodology, Tether combines local activity monitoring, retrieval-augmented generation (RAG), and gamification to offer real-time focus support and personalized dialogue. The system integrates operating system level system tracking to prompt engagement and its chatbot leverages ADHD-specific resources to offer relevant responses. Preliminary validation through self-use revealed improved contextual accuracy following iterative prompt refinements and RAG enhancements. Tether differentiates itself from generic tools by being adaptable and aligned with software-specific workflows and ADHD-related challenges. While not yet evaluated by target users, this work lays the foundation for future neurodiversity-aware tools in SE and highlights the potential of LLMs as personalized support systems for underrepresented cognitive needs.

Index Terms—ADHD, Assistive Tools, LLMs

I. INTRODUCTION

Equity, diversity, and inclusion (EDI) are increasingly recognized as critical in software engineering (SE), yet the field continues to lack representation across multiple identity dimensions [1], [2]. Although diversity is associated with innovation and improved team performance, the SE workforce remains predominantly homogeneous [3], [4]. Most EDI-focused research in SE is gender-centered, with much less attention to race, nationality, age, or disability [2], [5]–[7]. This limited scope contributes to inequities in hiring, participation, and in the design of tools and systems that often overlook the needs of diverse users [3], [6], [8]. Structural barriers—including implicit bias, intersectionality, and cultural taxation—further marginalize underrepresented developers and lead to higher attrition rates [1], [3], [6]. Addressing these challenges requires expanding the focus of EDI research and adopting inclusive practices that reflect the global population [2], [4].

Building on broader concerns around EDI in SE, recent studies have drawn attention to the specific challenges encountered by developers with ADHD, who represent over 10% of the industry [9], [10]. Despite their significant presence,

they often face persistent obstacles related to sustained attention, task initiation and completion, time management, and executive functioning [3], [10], [10]–[12]. These difficulties are compounded by workplace norms such as synchronous communication, constant context switching, and open office environments, which create additional cognitive strain [3], [8], [10], [12]. While many rely on coping mechanisms like timeboxing, external reminders, and rigid task structuring, these are often insufficient without broader organizational support [3], [10], [10], [12]. Formal accommodations remain underutilized due to stigma, lack of awareness, or unclear processes [1], [3], [10]. As a result, ADHD developers frequently report underperformance, stress, and job dissatisfaction [3], [10], reflecting a broader neglect of neurodiversity in SE and reinforcing the need for targeted research and inclusive practices [1], [8], [10].

In response to these gaps, recent studies across clinical and human-computer interaction domains have begun exploring the potential of Large Language Models (LLMs) as assistive tools for individuals with ADHD, offering novel approaches to address long-standing challenges in attention regulation and emotional support [13]–[16]. For example, LLM-powered conversational systems have been shown to improve user focus, emotional regulation, and overall engagement through interactive virtual characters tailored to ADHD users' cognitive and emotional needs [13], [17], [18]. These systems incorporate features such as proactive conversation initiation, role-switching, and adaptive feedback to maintain user attention and reduce the cognitive burden of task initiation [17]–[19]. Other investigations into LLM-based interventions, such as those employing ChatGPT for therapy enhancement, report promising results in terms of empathy, communication adaptability, and expanded access to support services, particularly in under-resourced settings [13]–[16].

Considering that **existing LLM applications for ADHD support have primarily targeted general therapy or daily life assistance**, there remains a gap in addressing the specific cognitive demands faced by neurodivergent professionals in high-focus domains such as software development [14], [15], [17]. This gap leaves unaddressed the challenges of managing focus and maintaining task flow in technical work environments. To explore this problem, we propose the following research question: **RQ. How can an LLM-powered tool**

support software engineers with ADHD in structuring their workflows and maintaining focus during development tasks? This paper presents preliminary results of the design and evaluation of our novel approach, offering a new direction for assistive technologies that better reflect the lived realities of neurodivergent professionals in SE [10], [11].

II. BACKGROUND

A growing number of digital tools have been developed to support focus and attention in individuals with ADHD, offering a foundation for technology-driven interventions. These tools address core challenges such as distractibility, poor sustained attention, and difficulty managing tasks [20]–[22], while also supporting executive functioning, academic development, emotional regulation, and motivation [23], [24]. Clinical research [25] highlights that some of the effective interventions for ADHD at an adult age consist of externalizing higher-level executive processes (e.g., planning, organization, inhibition, time management) by embedding support into the environment rather than relying solely on internal control. This principle is echoed in evidence-based interventions that leverage structured routines, reminders, and contextual scaffolding to reduce cognitive load and support behavior regulation.

Educational apps like Say-it and Learn use music, animation, and real-time feedback to teach literacy and math through interactive methods [26], [27], and Supangan offers gamified lessons with audio feedback to reinforce participation and concept retention [21]. For self-regulation and routines, tools such as TangiPlan guide morning activities, ChillFish promotes calm breathing through biofeedback, and BlurtLine provides tactile feedback to reduce impulsive speech [26]. Cognitive training is supported by the ADHD Trainer (Tajima Cognitive Treatment), which targets attention and memory with measurable improvements in task performance [26], [28].

A subset of these tools specifically focuses on attention management and distraction reduction, which are central concerns for individuals with ADHD. Apps like Stayfocusd and Leechblock restrict access to distracting websites, while SimplyNoise uses ambient sound to support concentration [26]. The N-back app provides cognitive training for working memory and sustained attention [26]. Living Smart assists adults in structuring routines and reducing disorganization, and Snappy uses motion data to help users monitor impulsivity and attention [23]. CASTT is designed to detect when learners lose attention and provide prompts to help them refocus in classroom settings [26]. Altogether, these eight applications (i.e., Stayfocusd, Leechblock, SimplyNoise, ADHD Trainer, Living Smart, N-back, CASTT, and Snappy) highlight the range of approaches used to enhance attention regulation and focus among individuals with ADHD across different age groups and contexts.

III. METHOD

Following ADHD treatment principles catalogued by Fleming and McMahon [25], Tether aims to externalize key executive processes (e.g., sustained attention, task initiation, and

planning) through LLM-driven scaffolding. By embedding these supports into the developer's real-time work environment, the tool helps mitigate the “double-deficit” in self-regulation characteristic of emerging adults with ADHD. This study adopts the Engineering Research methodology [29], a scientific method that focuses on the design and evaluation of software artifacts to address real-world challenges. Our artifact is a conversational desktop assistant intended to support software engineers with ADHD in managing focus and task engagement. The methodology emphasizes iterative development, user-centered evaluation, and transparency of design decisions to support reproducibility.

A. Tool Development

Tether was developed as a desktop application to ensure privacy-preserving local execution, minimize disruptions, and offer real-time, adaptive support for ADHD-related challenges, drawing on prior work in software and assistive technologies for ADHD [20]–[23], [26], [28]. Its architecture is shaped by four key components: a) a **bot interface** that delivers natural, adaptive interactions through a conversational assistant, helping reduce cognitive overhead and improve engagement for users with attentional difficulties [30], [31]; b) **LLM support**, which enables context-aware emotional and technical support without requiring structured input [32], [33]; c) a **retrieval-augmented generation (RAG)** pipeline built with LangChain, enhancing factual grounding and personalization by indexing ADHD-related literature and prior user interactions [34], [35]; and d) a lightweight **gamification engine** that tracks focus-related behaviors and rewards users with points, badges, and interface customizations to reinforce engagement in a non-disruptive way [21], [22], [24]. Tether also integrates a local monitoring engine that collects data on active window usage, idle time, and recovery patterns. These signals are used to generate structured prompts that trigger chat interactions and notifications. All components, including the Flask API backend, the local SQLite chat history, and the notification engine, are designed to operate locally, except for external LLM calls.

B. Preliminary Evaluation

In this preliminary stage, the evaluation was conducted based on a comparison with existing tools for focus and attention support described in the literature (see Section II). This comparison considered the design scope, interaction models, and contextual adaptability of each tool, guided by common ADHD-related challenges such as task initiation and attention regulation. No formal user study was conducted during this stage. This validation strategy aligns with engineering research practices that emphasize early, utility-oriented iteration as part of artifact development [29].

IV. PRELIMINARY RESULTS

Tether is a desktop application developed to support software engineers with ADHD by combining local activity monitoring, contextual prompting, and structured feedback mechanisms. The application is built using a modular architecture

consisting of a frontend interface built with Electron and React, and a backend served through a Flask REST API as seen in Figure 1, and it is available at: <https://github.com/SeallLab/Tether/releases/tag/v0.3.0>. A local SQLite database is used to store chat history. The system architecture is designed to run entirely on the user's machine to preserve privacy, with the exception of calls made to any LLMs. Tether includes integrations with native operating system services, including active window tracking, idle state detection, and system notifications. These services continuously record usage patterns and behavioral signals, which are used as context inputs for other components of the system. A retrieval-augmented generation pipeline, implemented using LangChain, indexes ADHD-specific reference materials and previous user interactions to support contextual response generation. The system uses Gemini for language generation and Gemini's embedding model for document retrieval. Figure 1 provides a high-level overview of the system architecture. Due to the conference page limit, additional implementation details are available here: <https://figshare.com/s/9f5e8e201fdf0b26fc36>.

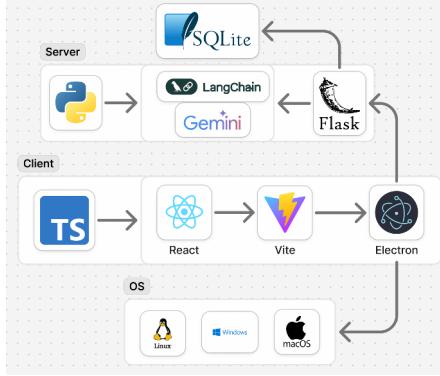


Fig. 1. System architecture of the Tether application.

A. User Workflow

Tether is designed to help software engineers with ADHD stay focused and re-engage when attention drifts. When users step away from their work or stop interacting with their computer for a prolonged period, the tool detects this inactivity and sends a supportive notification. These gentle nudges are tailored based on recent activity, such as which apps were in use or how long the user has been idle. Rather than interrupting users abruptly, Tether delivers encouraging, personalized messages that help them return to their tasks without pressure (see Figure 2).



Fig. 2. Sample notification for prolonged loss in focus.

In addition to notifications, users can interact with Tether through a built-in chatbot designed to support both emotional regulation and task management. For example, if a user is feeling overwhelmed or stuck on a coding task, they can ask the chatbot for help. The chatbot offers reflective prompts, grounding techniques, and software task guidance based on both recent activity and ADHD-specific reference material. It can suggest small, manageable steps, provide time boxing strategies, or simply offer empathetic conversation. This makes Tether a focus assistant and also a companion that understands the cognitive and emotional challenges (see Figure 3).

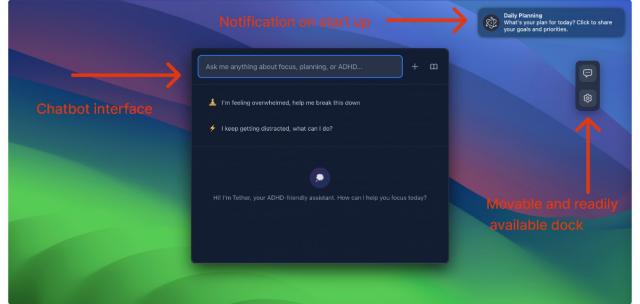


Fig. 3. Main user interface and chatbot elements.

To further support motivation and sustained engagement, a gamified progress tracker is included. As users stay focused, switch tasks efficiently, or recover quickly from distractions, they earn points, badges, and milestone rewards. These achievements are displayed in a dedicated interface and can be used to unlock alternate user interface themes. When users hit focus goals or complete challenges, Tether provides real-time positive reinforcement through notifications. This gamification layer is designed to reward effort and celebrate small wins, key motivators for users with ADHD (see Figure 4).

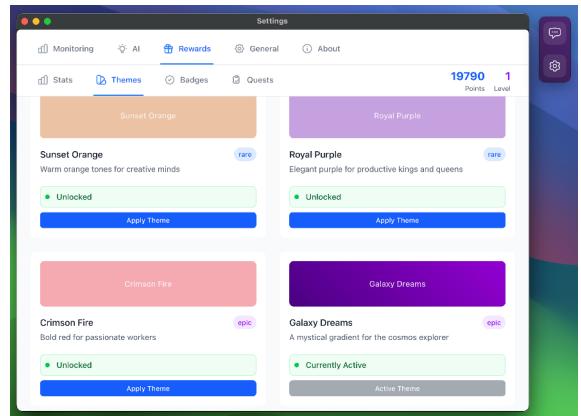


Fig. 4. Gamification aspects.

B. Internal Process

Tether's backend processes are initiated either through passive detection (e.g., user inactivity) or active engagement

(e.g., user interacting with the chatbot). The frontend tracks desktop activity, such as mouse movement, keystrokes, and app usage, and sends this data to a Flask backend, which analyzes it for signs of disengagement or context loss. If a trigger is detected, a structured prompt is generated using summaries of recent activity, chat history, and metadata. This prompt follows a modular template that incorporates ADHD-informed principles and guides the LLM's response type (e.g., motivational nudge, task suggestion, or emotional check-in). The system uses a retrieval-augmented generation (RAG) setup via LangChain, indexing both ADHD literature (e.g., the systematic literature reviews cited in Section 2) and user-specific data. The fully constructed prompt is then sent to Gemini's LLM API. Depending on the context and prompt instructions, the generated response is routed either to the native OS notification service (for nudges) or to the chatbot interface (for conversational support). The result is a responsive, personalized assistant grounded in real-time behavior and therapeutic strategies.

C. Preliminary Evaluation Results

As a preliminary evaluation, we compared Tether with tools available in the literature that were developed to support individuals with ADHD across behavioral and cognitive domains [20], [21], [23], [26], [28]. The comparison focused on differences in technology, domain, and focus area. In terms of technology, most existing tools are implemented as browser extensions (e.g., Stayfocusd, Leechblock), mobile apps (e.g., ADHD Trainer, N-back), or wearable devices (e.g., ChillFish). In contrast, Tether is a desktop application powered by LLMs that prioritizes local execution. Regarding the domain, prior tools are designed for general audiences or therapeutic support, whereas Tether specifically targets professional software engineers. Other tools focus areas include distraction blocking, cognitive training, and emotional regulation, but none offer support embedded within software development workflows. In addition to these dimensions, we explored whether each tool supports engagement, interaction, and gamification. This feature-level comparison is presented in Table I.

TABLE I
COMPARISON OF TETHER AND EXISTING ADHD-RELATED TOOLS

Tool	Monitoring	Chat	Dev-Aware	RAG	Gamified
Stayfocusd	No	No	No	No	No
Leechblock	No	No	No	No	No
SimplyNoise	No	No	No	No	No
ADHD Trainer	No	No	No	No	No
N-back	No	No	No	No	No
Living Smart	No	No	No	No	No
ChillFish	Yes	No	No	No	No
TangiPlan	No	No	No	No	No
Supangan	No	No	No	No	Yes
Say-it & Learn	No	No	No	No	Yes
Tether	Yes	Yes	Yes	Yes	Yes

V. DISCUSSION

Neurodivergence in software engineering has received growing attention in recent years, yet practical tools designed to support neurodivergent professionals, particularly those with ADHD, remain scarce. Tether tries to address this gap by

introducing a context-aware assistant that integrates directly into development workflows, offering real-time, adaptive support for attention management, task initiation, and emotional monitoring. Unlike existing tools focused on general behavior management or cognitive training, Tether responds to work dynamics and uses retrieval-augmented generation to generate timely, ADHD-informed prompts delivered via passive notifications or conversational chatbot interactions. Hence, the implications of this work are twofold. For research, Tether introduces a new direction for studying how assistive technologies can enhance neuroinclusive practices in software engineering, enabling investigations grounded in real-world behavioral data and LLM-driven interaction. For industry practice, our tool offers a deployable, lightweight support mechanism that integrates into existing tooling and rhythms of software development, reducing friction and enabling software engineers with ADHD to engage more fully and sustainably in their work.

VI. THREATS TO VALIDITY

Following the engineering research guidelines [29], relevant threats to validity must be acknowledged. This preliminary version of our tool was tested in a simulated setting with predefined scenarios, while it was being developed, which may not fully capture the needs and behaviors of professionals with ADHD in real-world development environments. In this study, our results were not validated with healthcare professionals specializing in ADHD, and no direct testing was conducted with neurodivergent software practitioners. These limitations point to the need for expert validation to assess the tool's effectiveness and relevance. Still, this work offers an important starting point for integrating neurodiversity-aware support into developer tools.

VII. FUTURE WORK

This paper presents early-stage results based on a working prototype of Tether, focusing on its design and internal validation for supporting software developers with ADHD. Future work includes extending the tool with additional sensing channels, such as camera-based gaze detection and microphone input for ambient noise, to better distinguish between focused inactivity and disengagement. We are also exploring ways to improve the chatbot's task-specific support by tailoring suggestions to coding and testing activities. For real-world validation, we are preparing two studies: one with healthcare professionals to assess therapeutic alignment and another with software engineers with ADHD to evaluate Tether in daily workflows.

VIII. CONCLUSIONS

To conclude, this paper introduced Tether, an early-stage prototype designed to support software engineers with ADHD using context-aware prompts, an adaptive chatbot, and gamified feedback. Preliminary results show promise for providing personalized, low-disruption support in real-world workflows, with potential to enhance neurodiversity inclusion. However, further development and validation are needed.

REFERENCES

- [1] R. d. S. Santos, C. Magalhaes, R. Santos, and J. Correia-Neto, "Exploring hybrid work realities: A case study with software professionals from underrepresented groups," in *Companion Proceedings of the 32nd ACM International Conference on the Foundations of Software Engineering*, 2024, pp. 27–37.
- [2] G. Márquez, M. Pacheco, H. Astudillo, C. Taramasco, and E. Calvo, "Inclusion of individuals with autism spectrum disorder in software engineering," *Information and Software Technology*, p. 107434, 2024.
- [3] N. da Silva Menezes, T. Á. da Rocha, L. S. S. Camelo, and M. P. Mota, "“i felt pressured to give 100% all the time”: How are neurodivergent professionals being included in software development teams?" in *Simpósio Brasileiro de Sistemas de Informação (SBSI)*. SBC, 2025, pp. 525–534.
- [4] P. Verma, M. V. Cruz, and G. Liebel, "Differences between neurodivergent and neurotypical software engineers: Analyzing the 2022 stack overflow survey," *arXiv preprint arXiv:2506.03840*, 2025.
- [5] G. Rodríguez-Pérez, R. Nadri, and M. Nagappan, "Perceived diversity in software engineering: a systematic literature review," *Empirical Software Engineering*, vol. 26, pp. 1–38, 2021.
- [6] K. Albusays, P. Bjorn, L. Dabbish, D. Ford, E. Murphy-Hill, A. Serebrenik, and M.-A. Storey, "The diversity crisis in software development," *IEEE Software*, vol. 38, no. 2, pp. 19–25, 2021.
- [7] K. K. Silveira and R. Prikladnicki, "A systematic mapping study of diversity in software engineering: a perspective from the agile methodologies," in *2019 IEEE/ACM 12th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE)*. IEEE, 2019, pp. 7–10.
- [8] K. Newman, S. Snay, M. Endres, M. Parikh, and A. Begel, "“get me in the groove”: A mixed methods study on supporting ADHD professional programmers," in *2025 IEEE/ACM 47th Intl Conference on Software Engineering (ICSE)*. IEEE Computer Society, 2025, pp. 778–778.
- [9] Stack Overflow, "Stack overflow developer survey 2022," 2022, accessed: 2025-06-24. [Online]. Available: <https://survey.stackoverflow.co/2022/>
- [10] K. Gama, G. Liebel, M. Goulão, A. Lacerda, and C. Lacerda, "A socio-technical grounded theory on the effect of cognitive dysfunctions in the performance of software developers with ADHD and autism," in *2025 IEEE/ACM 47th International Conference on Software Engineering: Software Engineering in Society (ICSE-SEIS)*. IEEE, 2025, pp. 1–12.
- [11] G. Liebel, N. Langlois, and K. Gama, "Challenges, strengths, and strategies of software engineers with adhd: A case study," in *Proceedings of the 46th International Conference on Software Engineering: Software Engineering in Society*, 2024, pp. 57–68.
- [12] M. R. Morris, A. Begel, and B. Wiedermann, "Understanding the challenges faced by neurodiverse software engineering employees: Towards a more inclusive and productive technical workforce," in *Proceedings of the 17th International ACM SIGACCESS Conference on computers & accessibility*, 2015, pp. 173–184.
- [13] S. Berrezueta-Guzman, M. Kandil, M.-L. Martín-Ruiz, I. Pau de la Cruz, and S. Krusche, "Future of ADHD care: Evaluating the efficacy of chatgpt in therapy enhancement," in *Healthcare*, vol. 12, no. 6. MDPI, 2024, p. 683.
- [14] S. Berrezueta-Guzman, M. Kandil, and S. Wagner, "Integrating ai into ADHD therapy: Insights from chatgpt-4o and robotic assistants," *Human-Centric Intelligent Systems*, pp. 1–16, 2025.
- [15] Y. Bennett, F. Gunturkun, M. Pillai, J. E. Herrmann, I. Luo, L. C. Huffman, and H. M. Feldman, "Applying large language models to assess quality of care: Monitoring ADHD medication side effects," *Pediatrics*, vol. 155, no. 1, p. e2024067223, 2025.
- [16] M. R. Olsen, C. Casado-Lumbreras, and R. Colomo-Palacios, "ADHD in ehealth-a systematic literature review," *Procedia Computer Science*, vol. 100, pp. 207–214, 2016.
- [17] T. Li, X. Hu, and X. Xu, "Design and evaluation of llm-based conversational virtual characters to assist adults with ADHD," in *Intl Conference on Human-Computer Interaction*. Springer, 2025, pp. 360–376.
- [18] B. Carik, K. Ping, X. Ding, and E. H. Rho, "Exploring large language models through a neurodivergent lens: Use, challenges, community-driven workarounds, and concerns," *Proceedings of the ACM on Human-Computer Interaction*, vol. 9, no. 1, pp. 1–28, 2025.
- [19] X. Wang, Q. Jia, L. Liang, W. Zhou, W. Yang, and J. Mu, "Artificial intelligence in ADHD: a global perspective on research hotspots, trends and clinical applications," *Frontiers in Human Neuroscience*, vol. 19, p. 1577585, 2025.
- [20] E. Kyriakaki and A. M. Driga, "Mobile applications for students with ADHD," *Global Journal of Engineering and Technology Advances*, vol. 15, no. 3, pp. 205–216, 2023.
- [21] A. Douloud, P. Pergantis, A. Drigas, and C. Skianis, "Managing ADHD symptoms in children through the use of various technology-driven serious games: A systematic review," *Multimodal Technologies and Interaction*, vol. 9, no. 1, p. 8, 2025.
- [22] C. R. Pásárelu, G. Andersson, and A. Dobreal, "Attention-deficit/hyperactivity disorder mobile apps: A systematic review," *International journal of medical informatics*, vol. 138, p. 104133, 2020.
- [23] L. Powell, J. Parker, V. Harpin *et al.*, "ADHD: is there an app for that? a suitability assessment of apps for the parents of children and young people with ADHD," *JMIR mHealth and uHealth*, vol. 5, no. 10, p. e7941, 2017.
- [24] J. Hernandez-Capistran, G. Alor-Hernandez, L. N. Sanchez-Morales, and I. Machorro-Cano, "A decade of apps for ADHD management: a scoping review," *Behaviour & Information Technology*, pp. 1–28, 2025.
- [25] A. P. Fleming and R. J. McMahon, "Developmental context and treatment principles for adhd among college students," *Clinical child and family psychology review*, vol. 15, pp. 303–329, 2012.
- [26] A. Douloud, A. Drigas, and C. Skianis, "Mobile applications as intervention tools for children with ADHD for a sustainable education," *Technium Sustainability*, vol. 2, no. 4, pp. 44–62, 2022.
- [27] S. Butt, F. E. Hannan, M. Rafiq, I. Hussain, C. N. Faisal, and W. Younas, "Say-it & learn: Interactive application for children with ADHD," in *Cross-Cultural Design. Applications in Health, Learning, Communication, and Creativity: 12th Intl Conference, CCD 2020, Held as Part of the 22nd HCI Intl Conference, HCII 2020, Copenhagen, Denmark, July 19–24, 2020, Proceedings, Part II 22*. Springer, 2020, pp. 213–223.
- [28] L. R. Carvalho, L. M. Haas, G. Zeni, M. M. Victor, S. P. Techele, J. M. Castanho, I. M. Coimbra, A. d. F. de Sousa, N. Ceretta, A. Garrudo *et al.*, "Evaluation of the effectiveness of the focus ADHD app in monitoring adults with attention-deficit/hyperactivity disorder," *European Psychiatry*, vol. 66, no. 1, p. e53, 2023.
- [29] P. Ralph, N. bin Ali, S. Baltes, D. Bianculli, J. Diaz, Y. Dittrich, N. Ernst, M. Felderer, R. Feldt, A. Filieri, B. B. N. de França, C. A. Furia, G. Gay, N. Gold, D. Graziotin, P. He, R. Hoda, N. Juristo, B. Kitchenham, V. Lenarduzzi, J. Martínez, J. Melegati, D. Mendez, T. Menzies, J. Moller, D. Pfahl, R. Robbes, D. Russo, N. Saarimäki, F. Sarro, D. Taibi, J. Siegmund, D. Spinellis, M. Staron, K. Stol, M.-A. Storey, D. Taibi, D. Tamburri, M. Torchiano, C. Treude, B. Turhan, X. Wang, and S. Vegas, "Empirical standards for software engineering research," 2020. [Online]. Available: <https://arxiv.org/abs/2010.03525>
- [30] S. Santhanam, T. Hecking, A. Schreiber, and S. Wagner, "Bots in software engineering: a systematic mapping study," *PeerJ Computer Science*, vol. 8, p. e866, Feb. 2022. [Online]. Available: <https://doi.org/10.7717/peerj.cs.866>
- [31] M. Wessel, M. A. Gerosa, and E. Shihab, "Software bots in software engineering: benefits and challenges," in *Proc. of the 19th International Conference on Mining Software Repositories*, ser. MSR '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 724–725. [Online]. Available: <https://doi.org/10.1145/3524842.3528533>
- [32] A. Fan, B. Gokkaya, M. Harman, M. Lyubarskiy, S. Sengupta, S. Yoo, and J. M. Zhang, "Large language models for software engineering: Survey and open problems," in *2023 IEEE/ACM International Conference on Software Engineering: Future of Software Engineering (ICSE-FoSE)*. IEEE, 2023, pp. 31–53.
- [33] J. Lin, X. Dai, Y. Xi, W. Liu, B. Chen, H. Zhang, Y. Liu, C. Wu, X. Li, C. Zhu *et al.*, "How can recommender systems benefit from large language models: A survey," *ACM Transactions on Information Systems*, vol. 43, no. 2, pp. 1–47, 2025.
- [34] Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, J. Sun, and H. Wang, "Retrieval-augmented generation for large language models: A survey."
- [35] G. Dong, Y. Zhu, C. Zhang, Z. Wang, J.-R. Wen, and Z. Dou, "Understand what llm needs: Dual preference alignment for retrieval-augmented generation," in *Proceedings of the ACM on Web Conference 2025*, 2025, pp. 4206–4225.