AI4EUD: Empowering the End-User — Software Development with LLMs

Thomas Weber[1], Passant Elagroudy[2], Philippe Palanque[3] and Sven Mayer[1]

[1]LMU Munich, Munich, Germany
[2]German Research Center for Artificial Intelligence, Germany
[3]ICS-IRIT, Université Paul Sabatier – Toulouse III, Toulouse, France

Abstract

Software is essential across many domains, yet many users lack the ability to create or modify code, limiting their digital agency and creative potential. End-User Development (EUD) seeks to bridge this gap, but traditional approaches struggle to balance ease of use with the expressiveness of programming languages. Recent advances in Generative AI (GenAI), particularly Large Language Models, offer promising opportunities to simplify code creation with more natural interaction. This workshop brings together researchers and practitioners in EUD, AI-assisted development, and Human-Computer interaction to explore how we can utilize GenAI to give end-user developers tools that maintain the expressiveness of modern programming languages, are easy to use and approachable, and also allow end-users to create high-quality, reliable software without requiring extensive software engineering expertise. Through the contributions of the workshop participants and hands-on exploration of different paradigms and presentations, we will assess the status of knowledge and chart a path forward toward a widely accessible, natural way to create software. Ultimately, by broadening the people's capability for software development, we aim to increase digital literacy, agency, and participation.

Keywords

End User Development, Generative AI, Large Language Models,

1. Motivation

Software is ubiquitous and essential in many domains, including research and industry, as well as personal use, such as education and entertainment. While many people have the skills to use software, only a much smaller group has the higher level of digital literacy that allows them to create, adapt, or customize software [1, 2, 3]. End-user development (EUD) promises to empower people [4] by allowing them to create, change and deploy software so that it meets their specific needs [5]. However, given the complexity of programming arbitrary software, many existing EUD systems must strike a balance between ease of use and expressiveness of a programming language [6, 7]. In consequence, end-user development paradigms such as visual programming, low-code and no-code development, and programming by example are typically limited in their expressiveness or are targeted at specific domains, thus limiting the range of scenarios where they can be applied [8]. In addition, creating software that is robust, reliable, and usable even years later remains difficult and requires experience and an understanding of software engineering body of knowledge and practices.

The use of generative AI (GenAI), such as large language models (LLM), has the potential to address these issues and to simplify the writing of code for both professionals [9, 10] and non-experts [11, 12]. End users can use LLMs not only to generate code from natural language descriptions but also to ask for alternatives, explanations, and assistance in case of errors. Although this can make programming more accessible, going beyond simple examples, use cases still introduce a high level of complexity [12, 13], which can make long-term and productive adoption of EUD with LLMs challenging. In addition,

Joint Proceedings of IS-EUD 2025: 10th International Symposium on End-User Development, 16-18 June 2025, Munich, Germany.
thomas.weber@ifi.lmu.de (T. Weber); passant.elagroudy@gmail.com (P. Elagroudy); philippe.palanque@irit.fr (P. Palanque); info@sven-mayer.com (S. Mayer)
https://sven-mayer.com (S. Mayer)
0000-0002-6894-605X (T. Weber); 0000-0003-1419-7425 (P. Elagroudy); 0000-0002-5381-971X (P. Palanque); 0000-0001-5462-8782 (S. Mayer)

understanding, adapting, and debugging the generated code and turning it into usable, reliable software remains a challenge.

Still, given the recent advances in LLMs, we are clearly only at the beginning of improving both the expressiveness and ease of use of programming languages by means of GenAI. Combining code generation with existing approaches and more natural representations [13, 14, 15, 16, 17] can increase interest in EUD approaches, create new synergies with GenAI, and overall make their benefits accessible to people with low programming proficiency. In addition, these paradigmatic changes can allow for completely novel interfaces for EUD, representations for software, and development process that specifically target people with low programming proficiency.

To embrace this momentum, we plan to bring together experts on EUD and GenAI-supported software development to assess the status quo of the intersection of these areas and envision how they can collaborate in the future. To this end, we will explore how existing and novel paradigms in EUD can synergize with GenAI-like LLMs to make the development of reliable software more accessible and proliferate digital literacy in a larger part of society to democratize software development.

2. Target Audience

With this workshop, we are looking to bring together experts from the field of EUD with those experienced in the use of GenAI and particularly in the use of LLMs for software development. This also includes those who work on software development paradigms or presentations that, so far, have found limited adoption from end-users but, with the introduction of GenAI, have become viable alternatives, like formal description techniques. We also welcome those interested in the democratization of software development and programming skills and more global digital literacy and the impact that this can have on society.

Expected Attendance We aim towards a participation of 10–20 people for lively discussions both in person and online.

3. Before the Workshop

We will publish a call for submissions and position papers prior to the workshop. We expect submissions from 2–8 pages that outline current research approaches and visions for how EUD can be transformed using GenAI to make it both easy to use and expressive. We also encourage reports on experiences and results on how to bridge the gap between natural/vernacular (and thus imprecise) input to reliable, high-quality software using current trends and technologies.

We will advertise the workshop through its website and the organizers' broad social media network, where we will publish the timeline for submissions, reviews, and camera-ready deadlines. After accepting submissions, workshop attendees will have the opportunity to extend and refine their submissions for publication.

4. Plan To Publish Proceedings

We will publish the workshop proceedings on CEUR-WS.org. Moreover, accepted submissions will be available on the workshop website in the weeks before the workshop with the authors' consent.

5. Workshop Mode

We will conduct the workshop as a full-day synchronous hybrid event with participants on-site and online. During the workshop, we will first assess the state of knowledge and existing projects based on participant submissions and experiences. In the second part of the workshop, we will work in groups

to explore how well we can utilize AI for different specification approaches and discuss the design requirements for interaction and interfaces for non-programmers.

9:00 – 9:30: Introduction The organizers will welcome the workshop participants and introduce the theme and goals of the workshop.

9:30 – 10:30: Talks — Status Quo Participants get to introduce themselves, including their background, area of expertise, experiences, and challenges with EUD and AI-supported development.

10:30 – 11:00: Break

11:00 – 12:30: Talks — Existing Approaches and Design Concepts Participants have the opportunity to present ongoing research projects that tackle the challenges of allowing people with low programming proficiency to create reliable software and how End User Development can leverage AI to make the experience more natural and overall successful. We will contribute a series of design concepts from our own work and teaching. We will then discuss how software reliability is usually assessed and how AI-supported programming requires changes in practice.

12:30 – 14:00: Lunch Break

14:00 – 15:30: Interactive Exploration of AI-generated Specifications After the attendees have heard about various approaches to how EUD can be enhanced with AI in the first half, we will work in smaller groups to test how well current AI models can generate various forms of software representations and how this affects EUD. This will include directly generating code and various intermediate representations, e.g., formal specifications, state machines, logical descriptions, etc. We will collect the options from the contributions of the first half, along with formats that might not have been covered by the participants. The participants will then work in groups to explore the individual benefits and challenges of these representations and how they address the existing shortcomings of EUD. This way, participants get to experience the challenges of this process, which can open up new research questions and projects. We plan to mix groups to enhance collaboration between experts with a different focus. Online participants will work in breakout rooms. Participants will be encouraged not just to follow the straightforward process but also to test out the limits of different representations and models.

15:30 – 16:00: Break

16:00 – 17:30: Roadmap for Solutions and Closing After the participants have identified what works well and where the challenges remain, we will collect individual feedback. The goal is to determine where there are commonalities and generalizable issues that need to be addressed to make AI-supported specification-driven EUD viable. From this, we will discuss the immediate, mid- and long-term research challenges to allow people with low programming proficiency to create reliable, high-quality software.

6. Organizers

Thomas Weber is a PhD researcher at LMU Munich, Germany, working on the intersection of HCI, AI and Software Engineering. In his research, he investigates how advances in AI affect Software Engineering practices from a human-centered perspective, e.g., the behavior of developers and how to support them so that they have the best prospects for creating high-quality software.

Passant Elagroudy is a post-doctoral researcher at the intersection of innovation, as she works at the German Centre for Artificial Intelligence (DFKI) and RPTU Kaiserslautern University, Germany. With a Ph.D. in Computer Science from the University of Stuttgart, her work focuses on creating technologies that enhance human cognition. She is also from the management team coordinating Humane AI Net, one of the largest research and industrial consortia, advancing ethical AI solutions for everyday life.

Philippe Palanque is a professor of computer science at the University Toulouse 3 "Paul Sabatier" in Toulouse, France. Since the late 80s, he has been working on the development and application of formal description techniques for interactive systems. For more than 20 years, he has been working on automation and its integration in interactive systems [18] for multiple domains [19]. For instance, he was involved in the research network HALA! (Higher Automation Levels in Aviation) funded by SESAR programm which targeted at building the future European air traffic management system. The main driver of Philippe's research over the last 20 years [20] has been to address in an even way Usability, Safety, and Dependability [21] in order to build trustable safety critical interactive systems. As for conferences, he is a member of the program committee of conferences in these domains, such as SAFECOMP 2025 (44th Conference on Computer Safety, Reliability and Security) and EICS 2025 (17th annual Conference on Engineering Interactive Computing Systems).

Sven Mayer is an assistant professor of computer science at LMU Munich. His research sits at the intersection between HCI and AI, where he focuses on the next generation of computing systems. He designs, builds, and evaluates future AI-driven human-centered interfaces.

Acknowledgments

This research project is funded by the Bavarian Research Institute for Digital Transformation (bidt), an institute of the Bavarian Academy of Sciences and Humanities.

Declaration on Generative AI

The author(s) have not employed any Generative AI tools.

References

[1] A. Martin, Digital literacy for the third age: Sustaining identity in an uncertain world, ELearning Papers 12 (2009).

[2] A. Rea, Coding equity: Social justice and computer programming literacy education, IEEE Transactions on Professional Communication 65 (2022) 87–103. doi:10.1109/TPC.2022.3143965.

[3] A. Vee, Understanding computer programming as a literacy, Literacy in Composition Studies 1 (2013) 42–64.

[4] F. Paternò, End user development: Survey of an emerging field for empowering people, International Scholarly Research Notices 2013 (2013) 532659. doi:https://doi.org/10.1155/2013/532659.

[5] G. Fischer, D. Fogli, A. Piccinno, Revisiting and Broadening the Meta-Design Framework for End-User Development, Springer International Publishing, Cham, 2017, pp. 61–97. doi:10.1007/978-3-319-60291-2_4.

[6] A. Repenning, A. Ioannidou, What makes end-user development tick? 13 design guidelines, in: H. Lieberman, F. Paternò, V. Wulf (Eds.), End User Development, Human-Computer Interaction Series, Springer, 2006, pp. 51–85. doi:10.1007/1-4020-5386-X_4.

[7] G. Fischer, E. Giaccardi, Y. Ye, A. G. Sutcliffe, N. Mehandjiev, Meta-design: a manifesto for end-user development, Commun. ACM 47 (2004) 33–37. doi:10.1145/1015864.1015884.

[8] B. A. Myers, A. J. Ko, C. Scaffidi, S. Oney, Y. Yoon, K. Chang, M. B. Kery, T. J.-J. Li, Making End User Development More Natural, Springer International Publishing, Cham, 2017, pp. 1–22. doi:10.1007/978-3-319-60291-2_1.

[9] T. Weber, M. Brandmaier, A. Schmidt, S. Mayer, Significant productivity gains through programming with large language models, Proc. ACM Hum.-Comput. Interact. 8 (2024). doi:10.1145/3661145.

[10] X. Hou, Y. Zhao, Y. Liu, Z. Yang, K. Wang, L. Li, X. Luo, D. Lo, J. Grundy, H. Wang, Large language models for software engineering: A systematic literature review, 2024. doi:10.48550/arXiv.2308.10620. arXiv:2308.10620.

[11] T. Calò, L. De Russis, Leveraging large language models for end-user website generation, in: L. D. Spano, A. Schmidt, C. Santoro, S. Stumpf (Eds.), End-User Development, Springer Nature Switzerland, Cham, 2023, pp. 52–61.

[12] M. X. Liu, A. Sarkar, C. Negreanu, B. Zorn, J. Williams, N. Toronto, A. D. Gordon, "what it wants me to say": Bridging the abstraction gap between end-user programmers and code-generating large language models, in: Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems, CHI '23, Association for Computing Machinery, New York, NY, USA, 2023. doi:10.1145/3544548.3580817.

[13] T. Wu, E. Jiang, A. Donsbach, J. Gray, A. Molina, M. Terry, C. J. Cai, Promptchainer: Chaining large language model prompts through visual programming, in: Extended Abstracts of the 2022 CHI Conference on Human Factors in Computing Systems, CHI EA '22, Association for Computing Machinery, New York, NY, USA, 2022. URL: https://doi.org/10.1145/3491101.3519729. doi:10.1145/3491101.3519729.

[14] V. J. Hobbs, D. J. Pigott, Facilitating end user database development by working with users' natural representations of, Human Computer Interaction Development & Management (2001) 271.

[15] U. Wajid, A. Namoun, N. Mehandjiev, Alternative representations for end user composition of service-based systems, in: M. F. Costabile, Y. Dittrich, G. Fischer, A. Piccinno (Eds.), End-User Development, Springer Berlin Heidelberg, Berlin, Heidelberg, 2011, pp. 53–66.

[16] S. Oney, B. Myers, J. Brandt, Interstate: a language and environment for expressing interface behavior, in: Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology, UIST '14, Association for Computing Machinery, New York, NY, USA, 2014, p. 263–272. URL: https://doi.org/10.1145/2642918.2647358. doi:10.1145/2642918.2647358.

[17] L. Beurer-Kellner, M. Fischer, M. Vechev, Prompting is programming: A query language for large language models, Proc. ACM Program. Lang. 7 (2023). URL: https://doi.org/10.1145/3591300. doi:10.1145/3591300.

[18] C. Fayollas, C. Martinie, P. Palanque, Y. Deleris, J.-C. Fabre, D. Navarre, An approach for assessing the impact of dependability on usability: Application to interactive cockpits, in: 2014 Tenth European Dependable Computing Conference, 2014, pp. 198–209. doi:10.1109/EDCC.2014.17.

[19] A. Meschtscherjakov, M. Tscheligi, B. Pfleging, S. Sadeghian Borojeni, W. Ju, P. Palanque, A. Riener, B. Mutlu, A. L. Kun, Interacting with autonomous vehicles: Learning from other domains, in: Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems, CHI EA '18, Association for Computing Machinery, New York, NY, USA, 2018, p. 1–8. URL: https://doi.org/10.1145/3170427.3170614. doi:10.1145/3170427.3170614.

[20] R. Bernhaupt, M. Cronel, F. Manciet, C. Martinie, P. Palanque, Transparent automation for assessing and designing better interactions between operators and partly-autonomous interactive systems, in: Proceedings of the 5th International Conference on Application and Theory of Automation in Command and Control Systems, ATACCS '15, Association for Computing Machinery, New York, NY, USA, 2015, p. 129–139. URL: https://doi.org/10.1145/2899361.2899375. doi:10.1145/2899361.2899375.

[21] P. Palanque, S. Basnyat, R. Bernhaupt, R. Boring, C. Johnson, P. Johnson, Beyond usability for safety critical systems: how to be sure (safe, usable, reliable, and evolvable)?, in: CHI '07 Extended Abstracts on Human Factors in Computing Systems, CHI EA '07, Association for Computing Machinery, New York, NY, USA, 2007, p. 2133–2136. URL: https://doi.org/10.1145/1240866.1240966.

doi:10.1145/1240866.1240966.