

Non-Limiting Technology: An Initial Proposal of an Emerging Concept in Software Engineering

Tatiana Cartagena¹, Rafael Araújo Rodrigues¹

¹Laboratório de Tecnologia, Inovação e Impacto Social (TECISlab)
Porto Alegre – RS – Brasil

tatiana@tecislab.com

Abstract. *This paper presents the initial proposal of Non-Limiting Technology (NLT), an emerging concept in Software Engineering that seeks to rethink the role of technology in human development. Unlike paradigms such as accessibility, usability, and inclusive design, NLT is based on the principle that technology should not impose limits on individuals' cognitive, social, or ethical capacities, but rather enhance their growth. The paper discusses related theoretical foundations, defines the initial conceptual pillars of NLT, and points out future research directions. It represents a conceptual effort to invite the community to explore new metrics and practices for a more human-centered software engineering.*

Resumo. *Este artigo apresenta a proposta inicial da Tecnologia Não Limitante (TNL), um conceito emergente na Engenharia de Software que busca repensar o papel da tecnologia no desenvolvimento humano. Diferente de paradigmas como acessibilidade, usabilidade e design inclusivo, a TNL parte do princípio de que a tecnologia não deve impor limites às capacidades cognitivas, sociais e éticas dos indivíduos, mas sim potencializar seu crescimento. O artigo discute os fundamentos teóricos relacionados, define pilares conceituais iniciais da TNL e aponta caminhos futuros de pesquisa. Trata-se de um esforço conceitual para convidar a comunidade a explorar novas métricas e práticas para uma engenharia de software mais humana.*

1. Introduction

Since the 1960s, Software Engineering has sought to balance technical quality such as reliability, performance, and maintainability [Pressman 2011] with concerns related to the user experience, including usability, accessibility, and inclusive design [Rogers et al. 2013]. These advances have made the field's knowledge more efficient and accessible. However, despite initiatives to incorporate human and cognitive aspects, studies examining the long-term impact of technology on cognitive development remain scarce [Gunatilake et al. 2024, Fagerholm et al. 2022]. Although rooted in Software Engineering, this proposal also dialogues with adjacent domains such as Human–Computer Interaction (HCI), software quality, AI ethics, cognitive science, and digital well-being. In this broader view, NLT complements rather than replaces established approaches, emphasizing potential *long-term* human impacts that extend beyond immediate task performance.

Digital technologies can both expand and limit cognitive and social abilities. Research distinguishes between effects *with* technology (immediate performance gains) and effects *of* technology (lasting cognitive residues that remain after use) [SALOMON et al. 1991]. Technology use can act as a protective cognitive reserve but is also linked to digital dementia, depending on engagement type [Benge and Scullin 2025]. Recent reviews confirm that while digital tools promote collaboration and learning, they may also cause attentional overload, dependence, and cognitive or social atrophies when poorly designed [Barros 2024, Shanmugasundaram 2023]. In this context, we propose the concept of Non-Limiting Technology (NLT), a conceptual proposition within Software Engineering that encourages designing software that avoids cognitive constraints, reduces dependence, and expands human capabilities. This paper presents the initial proposal of NLT, outlining its distinctions (2), conceptual pillars (3), and research agenda (4). It invites reflection: *How can we build software that not only works but also does not limit human potential?*

2. Related Work

The Non-Limiting Technology (NLT) proposal is situated within Software Engineering, establishing a complementary dialogue with reference areas. Although these domains are well established, they still present a conceptual gap regarding how to prevent limitations on human development, especially in long-term cognitive aspects [Gunatilake et al. 2024, Fagerholm et al. 2022]. Accessibility focuses on removing barriers that prevent individuals with sensory, motor, or cognitive disabilities from using software, following guidelines such as the WCAG [W3C Web Accessibility Initiative 2018]. Its mission is to ensure equitable access; however, its scope does not include cumulative cognitive effects from long-term use.

Usability has enhanced human-computer interaction through efficiency and satisfaction metrics [Nielsen 1994], while inclusive and universal design extends this discussion by accommodating greater user diversity [Clarkson et al. 2013]. Software quality models [ISO/IEC 2011] organize key attributes such as reliability, security, and maintainability. Although usability appears in these models, the absence of principles addressing human impact or long-term cognitive non-limitation reveals a conceptual gap. Within this panorama, NLT complements these areas. While related domains focus on access, interaction, and technical attributes, NLT places at the forefront the premise of designing software that does not impose limits or atrophies on human potential. In summary, accessibility ensures access, usability optimizes interaction, and inclusive design broadens participation, whereas Non-Limiting Technology seeks to preserve and expand human capacities over time. By emphasizing the *effects of technology* rather than only the *effects with technology*, NLT adds a long-term developmental dimension still absent from current paradigms.

3. Proposal: Non-Limiting Technology (NLT)

This study proposes Non-Limiting Technology as a guiding principle in Software Engineering. Its central premise is that software should be conceived in a way that does not impose limits or induce atrophies in human cognitive development. Unlike paradigms such as accessibility, usability, and inclusive design, NLT adopts non-limitation as a core criterion for software creation, focusing on long-term impacts.

3.1. Initial Definition

NLT establishes that software design must, from its earliest stages, promote the preservation and expansion of users' cognitive, social, and ethical capacities. A non-limiting software system, therefore, is one that, beyond its technical function, avoids inducing dependence or loss of abilities while fostering user autonomy and continuous growth.

3.2. Conceptual Pillars

To ensure that NLT is not merely an abstraction, its initial pillars derive from gaps and evidence identified in the literature on technology, neuroscience, and cognition. These pillars constitute the conceptual dimensions that will guide the development of future design guidelines:

- **Cognitive:** Focuses on preventing excessive dependence and skill loss (deskilling), seeking to promote lasting technological effects on executive functions (such as attention, memory, and planning) rather than merely achieving short-term performance gains.
- **Ethical:** Aims to preserve user autonomy and privacy, ensuring that software-supported decisions remain subject to human questioning and revision. This prevents the risk of unreflective automation and the weakening of critical judgment. It also encompasses broader concerns related to transparency, responsibility, and the ethical impact of emerging technologies on human decision-making.
- **Social:** Actively seeks to promote inclusion and collaboration, preventing software from contributing to isolation, cognitive overload, or restricted access to diverse perspectives.
- **Long-Term Human Development:** Emphasizes lifelong learning and continuous growth, considering the cumulative effects of technology use that transfer beyond the immediate task context.

These four pillars do not replace or invalidate existing consolidated areas; rather, they add a complementary and essential dimension a central concern with the long-term effects of software on human cognitive and social development.

4. Discussion and Future Work

The introduction of Non-Limiting Technology (NLT) aims to establish a new perspective for designing software that enhances human development while preventing cognitive atrophies and limitations. To address the research question "*How can we build software that not only works but also does not limit human potential?*", this study adopts non-limitation as a guiding perspective for software construction, structuring four conceptual pillars (Cognitive, Ethical, Social, and Human Development) to guide this process. This approach stands out because, while areas such as accessibility, usability, and inclusive design focus primarily on access and immediate interaction, NLT emphasizes the need to consider the long-term cognitive, social, and ethical impacts of technology. It recognizes that software can act as either an amplifier or a constraint on human capabilities.

In summary, this theoretical essay contributes by outlining an emerging conceptual framework that invites the Software Engineering community to rethink the relationship between technology and human development. The next steps involve empirically refining

these conceptual pillars and advancing guidelines and evaluation methods grounded in the NLT perspective. By framing non-limitation as a design consideration, NLT opens new directions for theoretical discussion, methodological exploration, and interdisciplinary collaboration aimed at building technologies that sustain, rather than restrict, human potential over time.

References

- Barros, E. C. (2024). Understanding the influence of digital technology on human cognitive functions: A narrative review. *IBRO Neuroscience Reports*, 17:415–422.
- Benge, J. F. and Scullin, M. K. (2025). A meta-analysis of technology use and cognitive aging. *Nature Human Behaviour*, 9(7):1405–1419.
- Clarkson, J., Coleman, R., Hosking, I., and Waller, S. (2013). *Inclusive Design: Design for the Whole Population*. Springer.
- Fagerholm, F., Felderer, M., Fucci, D., Unterkalmsteiner, M., Marculescu, B., Martini, M., Tengberg, L. G. W., Feldt, R., Lehtelä, B., Nagyvárad, B., and Khattak, J. (2022). Cognition in software engineering: A taxonomy and survey of a half-century of research. *ACM Comput. Surv.*, 54(11s).
- Gunatilake, H., Grundy, J., Hoda, R., and Mueller, I. (2024). The impact of human aspects on the interactions between software developers and end-users in software engineering: A systematic literature review. *Information and Software Technology*, 173:107489.
- ISO/IEC (2011). ISO/IEC 25010:2011, Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – System and software quality models. International Standard.
- Nielsen, J. (1994). *Usability Engineering*. Morgan Kaufmann, San Francisco.
- Pressman, R. S. (2011). *Engenharia de software: uma abordagem profissional*. AMGH, Porto Alegre, 7 edition.
- Rogers, Y., Sharp, H., and Preece, J. (2013). *Design de interação: além da interação humano-computador*. Bookman, Porto Alegre, 3 edition. Tradução: Isabela Gasparini.
- SALOMON, G., PERKINS, D. N., and GLOBERSON, T. (1991). Partners in cognition: Extending human intelligence with intelligent technologies. *Educational Researcher*, 20(3):2–9.
- Shanmugasundaram (2023). The impact of digital technology, social media, and artificial intelligence on cognitive functions: A review. *Frontiers in Cognition*, 2.
- W3C Web Accessibility Initiative (2018). Web content accessibility guidelines (wcag) 2.1. World Wide Web Consortium (W3C) Recommendation.