Proceedings Paper:
Stimson, C.E. orcid.org/0000-0002-2871-0894 and Raper, R. (2024) Participatory AI: a method for integrating inclusive and ethical design considerations into autonomous system development. In: Nazmul Huda, M., Wang, M. and Kalganova, T., (eds.) Towards Autonomous Robotic Systems (TAROS 2024). 25th TAROS Conference 2024, 21-23 Aug 2024, London, United Kingdom. Lecture Notes in Computer Science, 15051 (1). Springer Nature Switzerland , pp. 144-154. ISBN 9783031720581

https://doi.org/10.1007/978-3-031-72059-8_13

Participatory AI: A Method for Integrating Inclusive and Ethical Design Considerations into Autonomous System Development

Stimson, Christina E.[1] and Raper, Rebecca[2]

[1] The University of Sheffield, Sheffield, UK, cestimson1@sheffield.ac.uk
[2] Cranfield University, Cranfield, UK, rebecca.raper@cranfield.ac.uk

Abstract: There has been significant work in the field of AI Ethics pertaining to how it might offer *guidelines* for developers to design, develop and deploy AI in an ethical way. Recently, the European Union's AI Act has introduced a risk-based regulation approach for AI system development. However, despite the additional requirements the AI Act places on developers to ensure that their systems are created with transparency, fairness, and accountability etc., there is no *formalised methodology* for how this might be achieved. Drawing on the history of collaborative and emancipatory technology design in Scandinavia, this paper proposes a software development methodology founded on the ethics and praxis-based principles of Participatory Design. Integrating this approach into the established 'Waterfall Method', it offers developers a practical way of embedding ethics in AI development, and to thereby satisfy the requirements imposed by the new regulations.

Keywords: Participatory Design, Operationalising Ethics, AI Ethics, Agile Development.

1 Introduction

Artificial Intelligence (AI) Ethics, and ethics pertaining to autonomous systems more generally, is receiving increased attention, owing to what are perceived as existential risks associated with AI (i.e., threats from an Artificial General Intelligence), and well-documented issues such as AI bias and threats to job security. This paper focuses on the ethics relating to the less existential threats associated with autonomous systems (though these might be relevant) and proposes a formalised methodology for software developers to follow, following the Scandinavian tradition of Participatory Design. Hence, we introduce 'Participatory-AI' a procedure that inherently prioritises the *genuine participation* of stakeholders (primarily, end users).

The paper starts with some background to the topic area: outlining why there is a need for a formalised methodology for embedding ethics in AI software development, tracing discussions surrounding the operationalisation of ethics, followed by describing what Participatory Design is in the context of its history, as a more inclusive way to design technology solutions in the past. A new methodology – *Participatory AI* – is

then proposed with an in-depth description, followed by a discussion surrounding its *authentic application,* uses, and limitations. Finally, the paper concludes that 'Participatory AI' is an appropriate methodology for software developers to follow to practically embed ethical considerations in their development processes.

2 Background

2.1. Operationalising Ethics

The European Union (EU) AI Act recently set out 'levels of risk' associated with different use cases for AI. The act enforces bans on 'high level risk AI' and suggests that regulation be put in place to manage lower-level risk AI. Although the approach has generally been positively received, there are some fears that taking a regulation-based approach could stifle innovation [1].

Another issue with the regulation-based approach to ethical AI development is that, although there are rules and guidelines for what AI systems should look like (or what is and is not permitted regarding types of AI System (such as *'Ethically Aligned Design'* [2])), these do not prescribe to developers *how* to create the systems so that they meet the criteria set by the standards. For example, although guidelines such as 'ensure transparency' [2] are set as one metric for ethical AI, there are minimal prescriptions for how to achieve AI transparency. The topic has often been debated by philosophers, but there is little in the way of a formalised methodology for developers to follow to ensure that systems are created *with transparency*.

In Raper and Coeckelbergh (2022) [3], it was argued that the AI landscape has a *methodological gap* because whilst there are guidelines for ethical development, there are no formalised procedures that align with the processes that are typically followed by engineers and developers in designing their autonomous systems. 'Agile' forms of 'The Waterfall Method' (a formal process for requirements elicitation, development and testing) are frequently used by IT departments to ensure the rigorous, fit-for-purpose design of their new systems. However, there is yet no equivalent formalised methodology for designing autonomous systems or ensuring that ethical integrity an intrinsic aspect of the development process.

Recently, the term 'operationalising ethics' [4] has also been used to denote the need to practically apply ethics to the design process. Again, although there has been significant discussion on the need for a practical way to embed ethics in AI development processes (see [5] and [6]), comparatively little work has been done to address this need.

Ethical by Design [7], the incorporation of ethics into the design process, is frequently suggested as needed to satisfy this operational gap. However, there have been few attempts to put this into practice. One approach that attempts to operationalise ethics at the design stage of the AI development process is put forth in [8] with the suggestion that developers consider Spheres of Technology Influence (i.e., who and what the new technology will in turn affect), when designing an AI system. For instance, with one sphere being social impact, designers should consider what social impact their new system will have. The spheres are useful to highlight what is at stake if an AI system were to be unethically developed, but it still does not give instructions on how to build the systems, so that they take these spheres of influence into account. For instance, as a developer, I might envisage my technology being adopted in a certain way, but how can I ensure that the way it is adopted is ethical? This question highlights the necessity of rules and methods for ethical AI development.

Value Sensitive Design [9] is another approach that also tries to embed ethics into the AI design process. It emphasises the need to consider the values of stakeholders when designing new AI or autonomous systems. For instance, supposing a service robot were suggested to be introduced into a care home setting, the values of the elderly residents might be considered to determine what priorities should drive the new system design. However, this approach does not account for when conflicting values (in this case, residents with different worldviews) are driving the new system design. How should the new system be designed when there are opposing values from the individuals who will be affected by the new system? Furthermore, it is not obvious how values can or should be applied to new technology design. For instance, suppose a resident values honesty above and beyond all. It seems that the new system should be designed to maintain this value for the resident, but what does this mean for the new system? Does it mean that the new system cannot be deceptive, or does it mean that the system must be honest in how it interacts with participants? Though individual values are important to consider, individual values alone do not seem sufficient to drive specific technological requirements.

2.2. Participatory Design

Participatory Design (referred to hereafter as PD) is a broad term that refers to any design process that includes the active, sustained – but not necessarily full – involvement of end users (that is, those who would be affected by the designed artefact). In other words, artefacts are designed with users, rather than for them (cf. User-Centered Design). PD is not defined by a specific set of rules or methods, but by a commitment to two core principles:

- Enabling all who would be affected by a product/service to have their voice heard, regardless of their ability to `speak the language of professional technology design' [10].

- A process of mutual learning for both designers and users can inform all participants' capacities to envisage future technologies and the practices in which they are embedded and serve to enable ordinary people to be able to define what they want from a design process [10].

With its roots in various social, political, and civil rights movements from the 1960s and 1970s, PD has always been both inherently and expressly political in its aims [10].

PD possesses an intrinsic ethical, specifically an emancipatory quality due to a variety of reasons, the most salient of which are the regional and international social contexts. In particular, the Scandinavian tradition champions ‘an unshakeable commitment to ensuring that those who will use information technologies play a critical role in their design' [10].

This core value can be traced back to the Frankfurt School's critical theory. A theory that is critical is ‘distinguished from a “traditional” theory according to a specific practical purpose: a theory is critical to the extent that it seeks human “emancipation from slavery", acts as a “liberating… influence", and works “to create a world which satisfies the needs and powers of" human beings"'[11]). A fundamental contention of critical theory is that human emancipation in all circumstances of oppression ‘cannot be accomplished apart from the interplay between philosophy and social science through interdisciplinary empirical social research' [11].

In light of the above, it is important to understand the emancipatory context in which PD evolved to see how authentically collaborative methodologies such as PD can ensure the development of ethical and inclusive AI that is fit-for-purpose.

In the face of management-driven technological change in Scandinavian workplaces, early PD practitioners made a conscious and hitherto unprecedented decision to uphold the interests of workers – those who would be directly affected by the new computerised systems being imposed – over those of company bosses [10]. Pernicious attempts at task automation and de-skilling employees across industries as a method of worker subjugation, in combination with wider societal changes and a political milieu unique to Scandinavia – namely, an unusually strong tradition of trade unionism – were instrumental to the evolution of PD as it is understood today [10].

As stated, the decision to side with marginalised communities is attributed in part to societal changes occurring at the time. Increases in citizen engagement at local levels in Western European countries, along with internationally seismic events such the Vietnam War, led to a paradigmatic transition in IT design [10]. This transition afforded an understanding that technology is deeply bound by the social and political contexts in which it is used, as opposed to formalised best practices (as identified by Suchman

[12]). This concept, termed situated practice [12], is particularly useful as it reasserts the necessity for PD (of AI) to be continually reflexive and, therefore, iterative, to produce outcomes that are relevant to end users' reality: an inherently ethical objective.

Collaborations between researchers and trade unionists, early PD projects such as 'NJMF' [14], 'DEMOS' [15], and 'DUE' [16] very much kick-started and came to define Scandinavian PD in the 1970s. Rather than using PD merely as a means to designing a better product or increasing worker productivity, they treated 'democratic participation and skill enhancement' as valid and desirable ends in and of themselves [17].

Such a commitment to social justice distinguishes PD from (at least, superficially) similar user-oriented methodologies because it ensures that all outputs meet the needs and desires of their target user base, as they are not only deeply involved throughout the process, but decide how much, when, and how they are involved. This nuanced form of agency – that is, genuine, as opposed to "full" participation (see 'Discussion' section) – constitutes the *authentic* use of collaborative methodologies such as PD. Authentic collaboration inevitably increases user acceptance and long-term adoption, both issues that loom large over the field of AI.

PD, therefore, offers an alternative approach: one that enables people to have agency and meaningful involvement in the development of such a transformative and far-reaching innovation as AI. This sustained, egalitarian influence would go at least some way to address and offset the risks and issues posed by it (raised at the beginning of this paper).

3 Methodology

As stated earlier, Information Technology (IT) departments historically followed a process known as *The Waterfall Method* in the creation of new IT systems [18]. The purpose behind using such a methodological approach is that it prescribes to developers how to develop a new business technology solution so that it not only does what is intended from a business perspective, but also satisfies the requirements of the Product Owner.
Figure 1 illustrates a typical example of The Waterfall Method, with the various 'steps' denoted by stages in the waterfall.

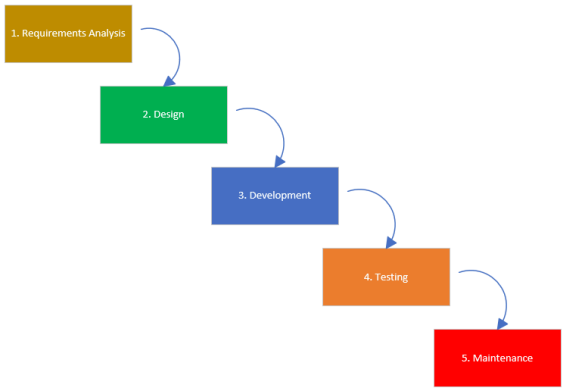

Fig 1. A typical representation of The Waterfall Method [19].

Per Figure 1, Step 1 of The Waterfall Method is **requirements analysis**. The aim of this stage is to capture the needs of the new system in as much detail as possible. For example, a new website designed to be accessible might need to have easy-read features. These requirements would be captured in the form of a matrix of logical requirements, with overarching aims at the top and further 'sub-requirements' under each aim. The overall aim of the project might be to create a website, but sub-requirements would specify how the website ought to look, feel, and operate.

After detailed requirements have been elicited, the Technical Architect specifies a **design** for the new system, forming Step 2 in The Waterfall Method. At this stage, there is close collaboration between the Business Analyst (the individual detailing the product requirements) and the Technical Architect to ensure that the proposed design specification meets the elicited requirements.

Step 3 is software **development.** Here, the requirements and technical architecture are developed and translated into a product using a preferred programming language. Once the product has been created, Step 4 takes place: the whole system is **tested** to ensure that the product meets the agreed specification. Using our example of a website that requires high accessibility, is it the case that the website is easy to read? Each listed requirement is mapped against a test criterion and checked against the final product.

Finally, once the system passes the testing phase, it is put into implementation, undergoing constant **maintenance** and periodic checks to ensure it still meets requirements and operates as intended.

Although The Waterfall Method is still very much in use across IT system development, Agile forms are more prevalent. Agile methodologies follow the same five steps listed above, except the steps take place at varying stages in the development process and are revisited if necessary. They take a non-linear, recursive, and iterative approach to software development. To manage these Agile approaches, a Product Owner is

appointed. Their responsibility is to ensure that the product adheres to the original design specification throughout the development process. As well as the traditional Project Manager, who ensures that the project is delivered on time, a 'SCRUM Master' is responsible for ensuring team and project cohesion.

Although attempts to involve users in technology design have been laudable and at least superficially fit-for-purpose, they are often extractivist in nature. In the context of late-stage capitalism, referring to the historical context described in the previous section, this means designers/developers look to exploit end users' insights to maximise profits, as opposed to meaningfully involve those who would be affected by their outputs. Examples of extractivist or otherwise not collaborative (that is, not geared towards increasing equity or management of power dynamics) include processes that only involve users at one or two stages. For example, only at an early ideation stage and/or a one-off confirmatory or amendments session. They are also often technocentric, or otherwise technologically-determinist. In such scenarios, the public is encouraged to 'view technological change as an inevitability and focus "on how to adapt to technology, not on how to shape it"' [13].

As stated earlier, Participatory Design is about ensuring that those affected by a new technology (primarily, end users) are involved in its design. As such, it is important to ensure that end users are involved in not necessarily all, but *most* stages of system development, and, crucially, *in a manner and to an extent of their choosing*.

We propose the following ways in which end users could be authentically involved in a new system design, integrated into The Waterfall Method as follows:

1. Requirements Analysis:

- Avoid 'educating' end users about the technology being developed. End users are to be considered domain experts (in other words, they know best what they need and want from a prospective product or service).
- Instead, try to foster mutual learning between the designers/developers/researchers and the (potential) end users in an equitable manner. This means getting to know each other's perspectives without pushing a particular agenda or focusing on generating a functionality checklist. Establishing a common parlance, for example co-creating UX personas or short stories, would be useful at this stage.
- The Product Owner is not the arbiter of whether the developed product meets the requirement criteria; *the end users are*. Only those who would actually use the product in real life can truly determine whether the product is fit-for-purpose and of sufficient quality.
- A key feature of this step in the process, if involving PD methods, is to remember that the design should not be driven exclusively by requirements for the business. If designing a new webpage for example, broader considerations should be considered: why is a new webpage being designed in the

first place? What utility will it serve for the community and wider economy? Will it serve to benefit everyone in society in an equitable and fair way?

2. Design:

- As PD is, by definition, iterative, reflexive, and recursive, it is important at this stage to ensure that multiple design feedback sessions are incorporated, where end users can participate how and as much, or as little, as they want. One session at the beginning of the development cycle and/or one at the end is not nearly enough, and constitutes at best box-ticking, and at worst extractivism.
- Much like how the Product Owner is not the sole or ultimate authority in whether the developed product passes muster, the Business Analyst and the Technical Architect must defer to the end users regarding how well the design reflects the requirements. This is not to say that end users should take on their job responsibility or burden; it is a shift how these roles are interpreted; they are facilitators, as opposed to judges or administrators.

3. Development:

- Throughout development, prototypes should be iterated upon according to end user and other stakeholders' feedback in a collaborative, cyclical effort. Again, the key here is agency as opposed to burden or responsibility. If an individual or group of participants shows little or no interest in getting involved at one stage or another, respect their choice but do not forget to keep the lines of communication open at future stages.

4. Testing:

- QA Testers hold valuable knowledge regarding not only how a product "should" be used, but how it can be broken either by accident or through a deliberate series of actions. Alternating between sharing their knowledge with end users and observing how end users interact with the prototype/product is a valuable means of different stakeholders learning from and with each other, which fosters not only meaningful participation, but better outcomes and improved rates of user acceptance and long-term adoption.

5. Maintenance:

- Involving end users in periodic checks – if they so desire – is both useful to ensuring the product continues to meet requirements and to fostering an enduring sense of meaningful involvement in participants and that their input is valued.

As an example case study, consider the aforementioned objective of designing an accessible website. An authentic PD methodology for this would meaningfully involve the individuals affected by the designed artefact. In trying to make the website accessible, the stakeholders to whom it should be made accessible should first be identified through stakeholder mapping, and then actively involved throughout the development process in a manner and extent of their choosing at each stage.

During Step 1: Requirements Analysis, this would include appointing the affected individuals as Product Owners and listening to their stories to understand their needs and desires regarding the product. It is crucial that the requirements list generated is founded on the *self-reported* needs and desires of the potential end users, as opposed to being largely driven by the needs of the business.

Throughout Step 2: Design, these potential end users should be treated as the arbiters of whether the design outputs meet their needs. During Step 3: Development and Step 4: Testing, the focus is solely on satisfying the needs of those who will ultimately use the accessible website. Test frameworks should be designed with this in mind, as well as adhere to the PD principles of enabling a polyphony of stakeholder voices and mutual learning between designers and end users.

Finally, when assuring the new system operates as intended in Step 5: Maintenance, developers must ensure that the system is working, and will continue to work faithfully; that is, in the way that was specified by non-business-oriented stakeholders during the PD process.

4 Discussion

Participatory Design is not a new concept; however, this paper introduces a new formalised procedure for ensuring that the approach is embedded in AI development lifecycles. Fundamental to this new approach is the active participation of implicated individuals during the project lifecycle; through this, developers would become more attuned to designing new systems with ethics at the forefront.

However, as PD is not new, there have been some issues with its implementation, mainly a lack of authenticity. As already stated, this means the genuine, but not necessarily full, involvement of end users. PD is often applied without embodying this quality. The following paragraphs outline some common pitfalls in this area.

Regarding participation, it is sometimes ambiguous as to whether any thought has been given to maintaining an equitable distribution or ratio of participants in any given session (for example, researchers/designers/developers to participants/end users, adults to children, or men to women). For example, in the context of Participatory Design in robotics research, papers often give the number of participants in each session along

with any salient demographic information, such as their sex, age, status/role, and state if any expected participants were absent from the session. However, whilst most provide an underlying logic or motivation for recruiting a particular number and/or demographic of participants to the study overall, there does not seem to be any adjustments made for when absences occur.

For example, in Šabanović et al. (2015) [19], the second PD workshop had three participants (two males and one female) and five researchers (sex distribution not stated, although can potentially be inferred from the paper's five named authors) in attendance, with two absentees (sex distribution not stated). This configuration is unequal, both with regard to sex distribution amongst the participants and the ratio of researchers to participants (Stimson, Roy & Szollosy, 2024, in preparation [20]).

Perhaps an in-situ decision to reduce the number of researchers present at the session might have bolstered the participants' feelings of being an equal partner in a conversation or endeavour, as opposed to a research subject being monitored for their responses. The question of whether excluding one male participant from said workshop to establish equal sex representation within it is an open and thorny one. There is the issue of fairness (indeed, which male participant out of the two should be excluded?). There is also whether the choice to exclude an available participant might undermine the depth and variety of the session's outcomes and insights, which would in turn affect its value to the study at large and its ability to make well-evidenced claims.

It is specificity and prior, considered elaboration on methodological choices that needs to be incorporated into development processes for autonomous systems.

It might be assumed from both ideological and research perspectives that maximizing participant input is ideal; indeed, this desire to democratise and decentralise power within and across relationships, practices and contexts is intrinsic to Scandinavian Participatory Design and to co-design in general. However, rather than constantly striving for strong, ‘high levels' of participation [21], desirable levels should be informed by the nature of the individual activity at hand and its rational and experiential aims. Participation in PD sessions must be driven by participants' interest and ability in each activity - even at the expense of researcher/project aims.

The transferal of greater agency to participants *when they express a desire for it* is key in development processes, as people often appreciate structure in new or unfamiliar processes, particularly children; they ‘recognise the limits of their autonomy and desire adult input and support' (Morrow cited in [21]). It is worth noting that even adults appreciate the input and support of those they consider more informed than they are. What is essential is doing so in a way that avoids patronising or undermining participants and enables them to be involved in a way that is comfortable and useful to them.

For example, when working with children, activities being fun is of paramount importance for engagement. Furthermore, the equalising of power relations between

researcher/designer/developer and all participants as much as possible takes precedence over researcher/designer/developer aims because Participatory Design and co-design are inherently political, emancipatory processes.

5 Conclusions and Future Work

This paper has offered the theoretical and practical foundations for a formalised methodology for incorporating Participatory Design practices into traditional software development processes so that developers can produce inherently ethical AI. PD incorporates ethics into technology design owing to its history involving marginalised communities in technology design. Considering the need for an operationalisation of ethics in AI development, we propose that, due to its emancipatory roots, PD can be introduced into typical software development lifecycles, such as those utilising The Waterfall Method and Agile variants thereof. If developers are to fulfil the new regulations by ensuring that AI developments are made in an ethical manner, an effective way of doing this is to incorporate PD throughout the software development cycle.

Considering future work in this area, we believe that once a step-by-step procedure has been established to cement PD in software development processes, then appropriate regulation in the form of development standards and development requirements can follow. For instance, there might be a new 'ethical' standard introduced to developed products to demonstrate that affected individuals have been involved in their software design (that the new system is ethical according to PD principles). There might also be broader regulations put into place to ensure that PD *is always applied* in the context of AI development that has significant implications (that is 'high risk' according to the EU AI Act). There is more research that needs carrying out in terms of formalising the above methodologies to ensure that affected individual voices are used to meaningfully shape new technology development. We hope this paper will inspire future work in this area.

Furthermore, additional work includes aligning the involvement of affected individuals with top-down governance procedures to ensure that the end product is effective in the form of ethical guidelines set for the design of new autonomous systems. For example, is the new system transparent to the affected individuals? Is it fair? These governance requirements might not be immediately obvious to those involved in designing the new system but would need checking through external auditing retrospectively.

Acknowledgments. This work is part of Research for the Imagining Technologies for Disability Futures project, and this project was funded by The Wellcome Trust (214963/Z/18/Z, 214963/B/18/Z, and 214963/C/18/Z).

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. BBC News, 2024, 'Davos 2024: Can – and should – leaders aim to regulate AI directly?' < https://www.bbc.com/worklife/article/20240118-davos-2024-can-and-should-leaders-aim-to-regulate-ai-directly> [Last Accessed: 29/04/2024]
2. Shahriari, K. and Shahriari, M., 2017, July. IEEE standard review—Ethically aligned design: A vision for prioritizing human wellbeing with artificial intelligence and autonomous systems. In 2017 IEEE Canada International Humanitarian Technology Conference (IHTC) (pp. 197-201). IEEE.
3. Raper, R. and Coeckelbergh, M., 2022. EKIP: designing a practical framework for embedding ethics in AI software development. ETHICOMP 2022, p.267.
4. Solanki, P., Grundy, J. and Hussain, W., 2023. Operationalising ethics in artificial intelligence for healthcare: A framework for AI developers. AI and Ethics, 3(1), pp.223-240.
5. Morley, J., Elhalal, A., Garcia, F., Kinsey, L., Mökander, J. and Floridi, L., 2021. Ethics as a service: a pragmatic operationalisation of AI ethics. Minds and Machines, 31(2), pp.239-256.
6. Morley, J., Kinsey, L., Elhalal, A., Garcia, F., Ziosi, M. and Floridi, L., 2023. Operationalising AI ethics: barriers, enablers and next steps. AI & SOCIETY, pp.1-13.
7. Mulvenna, M., Boger, J. and Bond, R., 2017, September. Ethical by design: A manifesto. In Proceedings of the European Conference on Cognitive Ergonomics (pp. 51-54).
8. Peters, D., Vold, K., Robinson, D. and Calvo, R.A., 2020. Responsible AI—two frameworks for ethical design practice. IEEE Transactions on Technology and Society, 1(1), pp.34-47.
9. Friedman, B., 1996. Value-sensitive design. interactions, 3(6), pp.16-23.
10. Simonsen, J. & Robertson, T., 2012, 'Routledge international handbook of participatory design', Routledge International Handbook of Participatory Design (63070), 37-63.
11. Bohman, J., 2004. 'Critical Theory'.
12. Suchman, L. A. (1987), Plans and situated actions : the problem of human-machine communication, Cambridge University Press, Cambridge.
13. Šabanović, S. Robots in Society, Society in Robots. Int J of Soc Robotics 2, 439–450 (2010). https://doi.org/10.1007/s12369-010-0066-7
14. Nygaard, K. (1979): The 'Iron and Metal Project': Trade Union Participation, in Sandberg (1979), pp. 94-107.
15. Sandberg, Å. (ed.) (1979): Computers dividing man and Work Swedish Center for Working Life, Demos Project Report no 13, Utbildningsproduktion, Malmö.
16. DUE Project Group (1979): Project DUE: Democracy, Development, and EDP, in Sandberg (1979), pp. 122-130.
17. Ehn, P. (2017), `Scandinavian design: On participation and skill', Participatory Design: Principles and Practices pp. 41-77.
18. Senarath, U.S., 2021. Waterfall methodology, prototyping and agile development. Tech. Rep., pp.1-16.
19. Šabanović, S., Chang, W. L., Bennett, C. C., Piatt, J. A. & Hakken, D. (2015), `A robot of my own: Participatory design of socially assistive robots for independently living older adults diagnosed with depression', Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 9193, 104-114.
20. Stimson, C. E., Roy, K., Szollosy, M. (2024) The Use of Participatory Design in Robotics: A Scoping Review [Manuscript in preparation]
21. Allsop, M. J., Holt, R. J., Levesley, M. C. & Bhakta, B. (2010), `The engagement of children with disabilities in health-related technology design processes: identifying methodology.', Disability and rehabilitation. Assistive technology 5(1), 1-13.