Tikanga in Technology

Indigenous Software Development Methods: an Overview

Aaron Matenga,[1] Annika Hinze,[1] Nicholas Vanderschantz,[1] Alvin Yeo,[1] Te Taka Keegan,[1] Ivy Taia[1] and Natalie Kusabs[2]

[1] Software Engineering, University of Waikato, Hamilton, New Zealand; [2] Te Kotahi Research Institute, University of Waikato, Hamilton, New Zealand

Tikanga In Technology

Te Mata Punenga o Te Kotahi - Te Kotahi Research Institute

Te Whare Wānanga o Waikato - The University of Waikato

Tikanga in Technology

Acknowledgements

The authors wish to acknowledge the that this research was partially supported by Te Kotahi Research Institute. We thank Ivy Taia (Taia, 2025) and Aaron Matenga for sharing their insights and unpublished findings, which have contributed to the development of this paper. An extended version is in preparation for publication.

Published by
Tikanga In Technology

Te Mata Punenga o Te Kotahi - Te Kotahi Research Institute
Te Whare Wānanga o Waikato - The University of Waikato
Private Bag 3105
Hamilton 3240
New Zealand

Email: rangahau@waikato.ac.nz
ISBN/DOI: XXX

Preferred citation: Matenga, A., Hinze, A., Vanderschantz, N., Yeo, A., Keegan, T., Taia, I., & Kusabs, N. 2025. Indigenous Software Development Methods: an overview. Information sheet for Tikanga in Technology. Te Kotahi Research Institute, Hamilton, New Zealand.

Introduction

Outcomes from software development projects are heavily influenced by the software development methodologies, methods, and processes applied (Bourque and Fairley, 2014). Research on *indigenous software development methods* is gaining traction (Hinze et al. 2024, Yeo et al, 2022), especially in the last decade, but there is little awareness of the breadth of alternatives. This paper reviews indigenous software development methods and highlights their difference to established software development methods.

A *software development method* is a general framework for the specification, design, construction, test, and verification of software products. Most software development methods share the same main steps, but complete cycles of these steps at different speeds based on competing priorities. Those priorities are typically determined by both internal and external stakeholders. *Stakeholders* are any people or groups of people who are affected by a software development project. They might exist within the organisation creating the software (internal), or outside of the organisation (external).

To aid comparison, we created a generalised software development method to use as a baseline for analysing any indigenous software development methods. We refer to this baseline method as the Generalised Software Development Life Cycle (GSDLC) as shown in Figure 1.

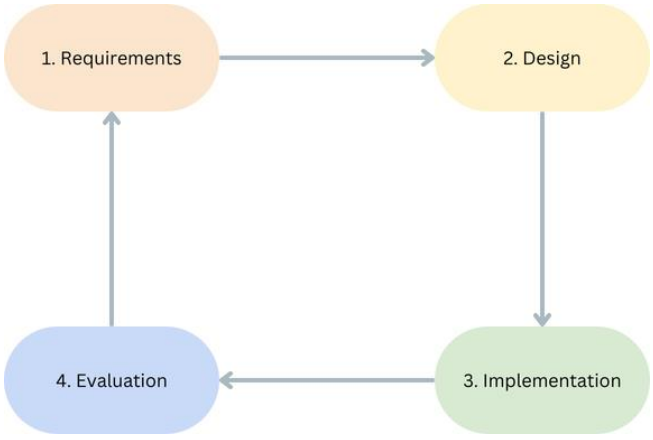

Fig. 1. Generalised software development life cycle.

Indigenous Software Development Methods

Recognising the value of culture and indigenous knowledge, several indigenous approaches to software development have emerged from research and industry. In some cases, the changes are small, such as using cultural terms for existing agile components (Sanabria et al., 2020). In this section we selected methods that offer something new or significantly different.

MCSE: The *Multicultural Software Engineering Working Model* (MCSE) is based on Boehm's Spiral method and recognizes knowledge creation throughout the entire software creation process. (Jaakkola et al., 2010). MCSE has three interacting main layers: Software engineering process layer, Knowledge context layer, and Multicultural context layer. This method is essentially a variant of the GSDLC with a focus on risk management. The MCSE does not offer any processes to accommodate cultural considerations but creates a mechanism for organisations to create their own knowledge around cultural impacts, and for that knowledge to inform their future development efforts.

PRISMA: *Participatory action Research In Software Methodology* Augmentation (PRISMA) is an amalgamation of Participatory Action Research and software development methodology (Siew et al., 2013). It has six main steps: a) problem exploration, b) hypothesis formation, c) design, d) tools & techniques, e) implementation and f) evaluation. These are grouped into the two overlapping processes: Social change (a, b, c, f) encompasses the change the community wants and the reasons they want it (closely related to steps 1, 2, and 4 of the GSDLC); Software development (c, d, e, f) covers the technical aspects (closely relate to steps 2, 3 and 4 of the GSDLC). PRISMA represents a second priority alongside that of developing software, which is that of social change. The method has been used to successfully build software for indigenous communities in Malaysia.

Tipi Ceremony as a Software Method: This method was used to create a mobile application for indigenous peoples, and is presented as seven steps (Banos, 2016): a) Birth offering and praying to the ancestors; b) Selecting the poles (i.e. core aspects to the software project); c) Placing the tripod (finding the right balance between competing priorities and engagement from various stakeholders); d) Putting the poles into position (creating the features of the mobile application); e) the canvas (interface and

performance); f) Crossbars and adjustments (feedback and adjustments made); g) the door and the lining (access code and privacy policy). This method is exploratory and would need more work before it could be generalised and applied to other software projects. It proposes both spiritual aspects and cultural processes as new ideas for incorporation into software methodologies and methods.

Mobile Development Approach: This approach builds on the Tipi Ceremony as a Software Method, and proposes four phases (Rizvi, 2022): In the preliminary phase, a proposal is created capturing the requests from clients or a community, and a Birth Ceremony is performed, led by an Elder. This is to signify that the digital platform has a spirit, just as the Tipi does. The Analysis phase matches the Design phase of the GSDLC, with particular attention to cultural considerations and policies. It also leans on sharing circles as the primary process for engaging with clients and users. Implementation and Maintenance closely match Steps 3 and 4 of the GSDLC.

Kaupapa Māori Co-design Process: A primary objective for this process was to enable Māori project partners to lead the conceptualisation, design, implementation and interpretation of research outcomes while developing a mobile health app (Te Morenga et al., 2018). This method has three main sections: research planning, design and implementation, and testing and evaluation. Although it was initially focused on research, it is also a close match to the GSDLC. Similar to the Tipi Ceremony, this process substitutes *hui* as an indigenous version of the focus group process. Notable differences to the GSDLC are the establishment of partnership values and Kaupapa Māori values which govern decisions and interactions throughout the entirety of the method.

9-Step Software Design Process: This process was developed as part of a co-design project between Māori and Tauiwi collaborators. It extends the GSDLC from four to nine steps (Rolleston et al., 2021) to support the full research project process. Three additional steps are added to the beginning (initial planning, preparation, team building). The initial planning covers preparing of the funding application and integrating Māori principles within the project plan and budget. The preparation step involved contract negotiations and defining clear outcomes and expectations. The team-building step revolved around establishing a shared vocabulary and understanding for the collaboration at an operational level. The requirements analysis is split into two steps acknowledge the engagement with end users and whānau for data collection, separate from the analysis of that data. It also incorporates the returning of collected data and reporting back of insights to participants. This shows respect as well as a desire for mutual benefits. The final step, ongoing relationships, recognises that relationships do not and should not cease at project completion.

Pathfinding: The Animikii company developed this software development approach for their work with indigenous peoples and marginalised communities (Animikii, 2025). The process includes a series of remote Pathfinding sessions and can be divided into six phases: a) defining the project (deliverables, contracts, initial deadlines, and finding alignment on how to move forward), b) discover the context, c) dream of the potential (needs, hopes, and goals of the users), d) design the interface, e) deliver the pathfinding report, f) budgeting the build. The pathfinding report helps determine the costs of project and can be used to apply for grants, as a business plan, for proof of concept, or even for another team to build the solution. Pathfinding aligns well with steps 1 and 2 of the GSDLC with focus on relationships as a prerequisite. It represents a shift in priority, valuing collaboration over competition.

SPEAR: This is conceptual framework for prioritising indigenous values, knowledge and capability development in the design and development of indigenous cultural games. It involves indigenous contributors as collaborators with a consideration for Sovereignty, Positionality, Equity, Advocacy, and Reciprocity through (LaPensee, 2020). SPEAR's values can be applied to all types of software development projects that aim to produce a solution for primarily indigenous users. However, SPEAR is focused on improving outcomes for indigenous peoples and may need some adjustments before being used in projects that do not involve any indigenous stakeholders.

Discussion and Conclusion

Each of the indigenous methods have approached software development with a set of values and principles that place emphasis on indigenous culture and knowledge. This pattern is different from and additional to the standard GSDLC (see Figure 2).

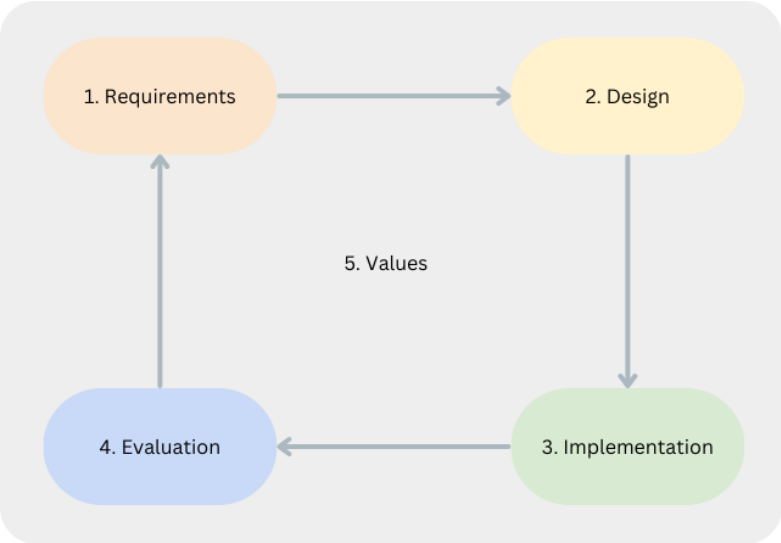

Fig. 2. Extended GSDLC acknowledging values

Most methods show an emphasis on increasing partner and user participation through a focus on building relationships. However, this participation does not typically extend to the technical implementation of the software. Table 1 shows a comparison of the indigenous methods to the extended GSDLC.

	SDLC Steps				
	1. Requirements	2. Design	3. Implementation	4. Evaluation	5. Values
MCSE	X	x	x	x	✓
PRISMA	✓	✓	x	✓	✓
Tipi	✓	✓	x	✓	✓
Kaupapa Māori	✓	✓	x	✓	✓
9-Step	✓	✓	x	✓	✓
Mobile Dev Approach	✓	✓	x	✓	✓
Pathfinding	✓	✓	x	✓	✓

Table 1. Indigenous methods compared to the extended GSDLC acknowledging values

We observe that none of the indigenous methods seem to consider the indigenous perspectives on the implementation step nor the views of indigenous project managers and developers. As the number of indigenous software developers continues to grow, we will address this opportunity for insights into their perspectives in our future research.

References

1. Animikii. Pathfinding: A different Approach to Software Development (at Animikii.com). https://animikii.com/insights/pathfinding-a-different-approach-to-software-development. Accessed 01/02/2025

2. A. M. Banos. Decolonizing technology through a Tipi: Creation of an Indigenous mobile application at York University. Master's thesis, York University, 2016.
3. P. Bourque and R. E. Fairley, editors. Guide to the Software Engineering Body of Knowledge. IEEE Computer Society, Los Alamitos, CA, version 3.0 edition, 2014.
4. H. Jaakkola, A. Heimburger, and P. Linna. Knowledge-oriented software engineering process in a multi-cultural context. Software Quality Journal, 18(2):299–319, June 2010.
5. A. Hinze, A. Yeo and D. Paruru. "Ethical Participatory Design: collaborative app development as indigenization of software engineering." In Proceedings of the Participatory Design Conference 2024: Exploratory Papers and Workshops-Volume 2, pp. 160-165. 2024.
6. E. LaPensee. SPEAR: a framework for indigenous cultural games. ANTARES: Letras e Humanidades, 12(28):4–22, 2020.
7. A. Rizvi. Design Approach for Building Technology with Indigenous Communities. Master's thesis, York University, 2022.
8. A. K. Rolleston, J. Bowen, A. Hinze, E. Korohina, and R. Matamua. Collaboration in research: weaving Kaupapa Māori and computer science. AlterNative: An International Journal of Indigenous Peoples, 17(4):469–479, 2021.
9. L. F. M. Sanabria, L. M. V. Ordonez, and E. A. Hoyos. Educational software for the initiation of the Kichwa language: SHIMISOFT. In Proceedings of the 5th Workshop on ICTs for improving Patients Rehabilitation Research Techniques, REHAB '19, pages 63–67, Jan. 2020.
10. S.-T. Siew, A. W. Yeo, and T. Zaman. Participatory action research in software development: indigenous knowledge management systems case study. In Proceedings of the 15th International Conference on Human-Computer Interaction, pages 470–479. Springer, 2013.
11. L. Te Morenga, C. Pekepo, C. Corrigan, L. Matoe, R. Mules, D. Goodwin, J. Dymus, M. Tunks, J. Grey, G. Humphrey, et al. Co-designing an mHealth tool in the New Zealand Māori community with a "Kaupapa Māori" approach. AlterNative: An International Journal of Indigenous Peoples, 14(1):90–99, 2018.
12. I.Taia Software Development: Involving Tangata Whenua and incorporating Tikanga Māori. Master's of Philosophy Thesis, University of Waikato, 2025 (in print)
13. A. Yeo, Hinze, A., Vanderschantz, N., Aporosa, S., & Paruru, D. Mobile app development: Work-integrated learning collaborations with Māori and Fijian partners. International Journal of Work-Integrated Learning, 23(2), 237-258. 2022