



Adaptabilitas *Waterfall* dan *Agile* dalam Pengembangan Perangkat Lunak: Sebuah Tinjauan Literatur Naratif

Muhammad Arfan

Program Studi Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam,
Universitas Halu Oleo, Kota Kendari, Indonesia

*Email Korespondensi: muhammadarfanbandi@uho.ac.id

Diterima: 16-09-2026 | Disetujui: 22-09-2026 | Diterbitkan: 24-09-2026

ABSTRACT

The selection of a software development methodology is an important aspect in determining how requirements, development processes, testing, and changes are managed throughout the software life cycle. Waterfall and Agile have different characteristics in terms of process structure, planning, requirements management, stakeholder involvement, and responsiveness to change. This study aims to analyze the characteristics, strengths, limitations, and adaptability of Waterfall and Agile based on the context of software development. The study employs a Narrative Literature Review (NLR) method by analyzing 22 sources of literature. The literature was narratively analyzed by grouping the findings according to methodological characteristics, application contexts, organizational and project factors, and hybrid approaches. The findings indicate that Waterfall is characterized by its suitability for projects with relatively stable requirements, well-defined scopes, and high documentation and regulatory demands, whereas Agile supports projects with evolving requirements, high uncertainty, and dynamic environments. The review also indicates that organizational culture, team competencies, project complexity, and regulatory requirements influence the suitability of a development methodology. Hybrid approaches have emerged as an alternative for integrating Waterfall and Agile practices according to project requirements. Therefore, the selection of a software development methodology should be conducted contextually based on project characteristics and the organization's ability to manage change.

Keywords: *adaptability; agile; NLR; waterfall*

ABSTRAK

Pemilihan metodologi pengembangan perangkat lunak merupakan aspek penting dalam menentukan bagaimana kebutuhan, proses pengembangan, pengujian, dan perubahan dikelola selama siklus hidup perangkat lunak. Waterfall dan Agile memiliki karakteristik yang berbeda dalam struktur proses, perencanaan, pengelolaan kebutuhan, keterlibatan pemangku kepentingan, serta kemampuan merespons perubahan. Penelitian ini bertujuan menganalisis karakteristik, kekuatan, keterbatasan, dan adaptabilitas Waterfall dan Agile berdasarkan konteks pengembangan perangkat lunak. Penelitian menggunakan metode *Narrative Literature Review* (NLR) dengan menganalisis 22 sumber literatur. Literatur dianalisis secara naratif dengan mengelompokkan temuan berdasarkan karakteristik metodologi, konteks penerapan, faktor organisasi dan proyek, serta pendekatan *hybrid*. Hasil kajian menunjukkan bahwa Waterfall memiliki karakteristik yang mendukung proyek dengan kebutuhan relatif stabil, ruang lingkup terdefinisi, dan tuntutan dokumentasi serta regulasi yang tinggi, sedangkan Agile mendukung proyek dengan kebutuhan yang berkembang, ketidakpastian tinggi, dan lingkungan dinamis. Kajian juga menunjukkan bahwa budaya organisasi, kompetensi tim, kompleksitas proyek, dan tuntutan regulasi memengaruhi kesesuaian metodologi. Pendekatan *hybrid* berkembang sebagai alternatif untuk mengintegrasikan praktik Waterfall dan Agile sesuai kebutuhan proyek. Dengan demikian, pemilihan metodologi perlu dilakukan secara kontekstual berdasarkan

karakteristik proyek dan kemampuan organisasi dalam mengelola perubahan.

Katakunci: *adaptabilitas; agile; NLR; waterfall*

Bagaimana Cara Sitasi Artikel ini:

Arfan, M. (2026). Adaptabilitas Waterfall dan Agile dalam Pengembangan Perangkat Lunak: Sebuah Tinjauan Literatur Naratif. *Journal of Literature Review*, 2(2), 1015-1026. <https://doi.org/10.63822/dbb1c810>

PENDAHULUAN

Pemilihan metodologi pengembangan perangkat lunak merupakan salah satu aspek penting dalam *software development* karena berpengaruh terhadap bagaimana kebutuhan, proses pengembangan, pengujian, pengelolaan perubahan, dan penyampaian produk perangkat lunak dilakukan. Dalam konteks *Information Systems Development* (ISD), pemilihan metodologi tidak hanya berkaitan dengan tahapan teknis pengembangan, tetapi juga perlu mempertimbangkan karakteristik proyek, struktur organisasi, kebutuhan pemangku kepentingan, serta dinamika lingkungan pengembangan (Saravanos & Curinga, 2023). Salah satu pendekatan tradisional yang memiliki pengaruh kuat dalam sejarah pengembangan perangkat lunak adalah model Waterfall. Model ini mengorganisasikan proses pengembangan ke dalam tahapan yang berlangsung secara berurutan, sehingga keluaran dari suatu tahap menjadi dasar bagi pelaksanaan tahap berikutnya (Beasley dkk., 2022). Struktur tersebut memberikan kerangka proses yang sistematis dan terdokumentasi, sehingga sesuai untuk lingkungan pengembangan yang memiliki kebutuhan relatif stabil serta memerlukan pengendalian proses yang terstruktur (Hassani & Idrissi, 2022). Berbeda dengan Waterfall, Agile berkembang sebagai pendekatan yang menekankan pengembangan secara iteratif dan inkremental dengan memberikan perhatian lebih besar terhadap kolaborasi, umpan balik pengguna, serta kemampuan merespons perubahan kebutuhan selama proses pengembangan (Pavličková dkk., 2024). Karakteristik tersebut memungkinkan tim pengembang memperoleh umpan balik secara lebih cepat dan melakukan penyesuaian terhadap produk berdasarkan perubahan kebutuhan maupun kondisi proyek (Dendere dkk., 2021). Penerapan prinsip Agile juga berkaitan dengan perubahan pola kerja organisasi yang semakin menekankan kolaborasi lintas fungsi, adaptabilitas, dan pengembangan yang berorientasi pada kebutuhan pengguna (Mergel dkk., 2021).

Perbedaan karakteristik antara Waterfall dan Agile menyebabkan kedua pendekatan tersebut memiliki implikasi yang berbeda terhadap pengelolaan proyek pengembangan perangkat lunak. Waterfall cenderung memberikan struktur proses, dokumentasi, dan tahapan pengembangan yang lebih terdefinisi, sedangkan Agile memberikan fleksibilitas yang lebih besar dalam menghadapi perubahan kebutuhan dan memungkinkan pengiriman produk secara bertahap (Hassani & Idrissi, 2022). Oleh karena itu, efektivitas suatu metodologi tidak dapat dilepaskan dari konteks penerapannya, termasuk stabilitas kebutuhan, kompleksitas proyek, karakteristik organisasi, serta tingkat perubahan lingkungan pengembangan (Mishra & Alzoubi, 2023). Perkembangan praktik pengembangan perangkat lunak selanjutnya menunjukkan bahwa penerapan metodologi tidak selalu mengikuti satu pendekatan secara murni. Dalam kondisi tertentu, organisasi dapat mengombinasikan praktik dari pendekatan tradisional dan Agile untuk menyesuaikan proses pengembangan dengan kebutuhan proyek dan karakteristik organisasi (Diansyah dkk., 2023). Pendekatan tersebut dikenal sebagai *hybrid software development*, yang mencerminkan upaya mengintegrasikan mekanisme perencanaan dan pengendalian yang terstruktur dengan praktik iteratif dan adaptif dalam satu proses pengembangan (Grinberg & Niță, 2022).

Berdasarkan perkembangan tersebut, kajian Waterfall dan Agile perlu diarahkan pada perspektif kontekstual yang mempertimbangkan karakteristik, keterbatasan, serta kemampuan masing-masing metodologi dalam beradaptasi terhadap kebutuhan proyek dan perubahan lingkungan pengembangan (Mishra & Alzoubi, 2023). Penelitian ini bertujuan untuk menganalisis adaptabilitas Waterfall dan Agile melalui kajian literatur naratif, dengan fokus pada karakteristik, penerapan, keterbatasan, dan faktor kontekstual yang memengaruhi penggunaannya dalam pengembangan perangkat lunak. Hasil kajian diharapkan memberikan pemahaman mengenai kondisi yang memengaruhi penerapan dan adaptasi kedua

metodologi dalam berbagai konteks pengembangan perangkat lunak.

METODE PENELITIAN

Penelitian ini menggunakan pendekatan *Narrative Literature Review* (NLR) untuk mengidentifikasi, mengkaji, dan mensintesis penelitian terdahulu mengenai karakteristik, kelebihan, keterbatasan, dan adaptabilitas metodologi Waterfall dan Agile dalam pengembangan perangkat lunak. Penelusuran literatur dilakukan melalui Wiley Online Library, Emerald Insight, SpringerLink, IEEE Xplore, MDPI, SAGE Journals, AIS eLibrary, dan CSIRO Publishing, dengan kata kunci “*Waterfall*”, “*Agile*”, “*Waterfall vs Agile*”, “*software development methodology*”, “*software development lifecycle*”, “*hybrid software development*”, “*project management methodology*”, dan “*methodology selection*”. Literatur yang dipilih berfokus pada publikasi yang membahas karakteristik, implementasi, perbandingan, adaptasi, dan konteks penerapan metodologi pengembangan perangkat lunak.

Seleksi literatur dilakukan berdasarkan kesesuaian dengan tujuan penelitian, yaitu publikasi yang membahas Waterfall, Agile, atau pendekatan hybrid dalam pengembangan perangkat lunak dan sistem informasi. Publikasi yang tidak berkaitan langsung dengan metodologi pengembangan atau tidak memberikan pembahasan substantif dikeluarkan dari kajian. Berdasarkan hasil seleksi, terdapat 22 sumber yang dipublikasikan pada periode 2020–2024 dan digunakan sebagai literatur akhir dalam kajian. Jumlah tersebut merupakan sumber yang dianalisis dan dikutip, bukan jumlah keseluruhan artikel yang diperoleh pada tahap awal pencarian.

Data dari literatur terpilih diekstraksi berdasarkan metodologi yang dibahas, karakteristik, kelebihan, keterbatasan, konteks penerapan, serta bentuk adaptasi terhadap perubahan kebutuhan dan kondisi proyek. Selanjutnya, data dikelompokkan ke dalam tema karakteristik Waterfall dan Agile, perbandingan kedua metodologi, adaptabilitas, pendekatan hybrid, dan faktor kontekstual. Hasil ekstraksi kemudian disintesis secara naratif untuk mengidentifikasi pola, perbedaan, dan kondisi yang memengaruhi penerapan serta adaptabilitas Waterfall dan Agile dalam berbagai konteks pengembangan perangkat lunak.

HASIL DAN PEMBAHASAN

Metode Pengembangan Waterfall

1. Karakteristik Waterfall

Model Waterfall merupakan salah satu model pengembangan perangkat lunak tradisional yang diperkenalkan melalui karya Winston W. Royce pada tahun 1970 dan menggunakan alur pengembangan yang berlangsung secara berurutan dari satu tahap ke tahap berikutnya (Pavličková dkk., 2024). Model ini umumnya mencakup tahapan analisis kebutuhan, perancangan, implementasi, pengujian, penerapan, dan pemeliharaan yang dilaksanakan secara sistematis (Saravanos & Curinga, 2023). Waterfall memiliki karakteristik *plan-driven*, yaitu kebutuhan, ruang lingkup, jadwal, sumber daya, dan keluaran proyek direncanakan sejak tahap awal pengembangan (Hassani & Idrissi, 2022). Setiap tahap menghasilkan dokumentasi tertentu yang menjadi dasar bagi tahap berikutnya sehingga proses pengembangan memiliki struktur, dokumentasi, dan keterlacakan yang relatif kuat (Diansyah dkk., 2023).

Pemisahan tahapan yang jelas memungkinkan pengelolaan proyek dilakukan melalui *milestone* dan keluaran yang terdefinisi. Karakteristik tersebut mendukung proyek yang memerlukan dokumentasi formal,

proses verifikasi dan validasi, serta kepatuhan terhadap standar atau regulasi tertentu (Thota & Jung, 2024). Dengan demikian, karakteristik Waterfall menunjukkan orientasi yang kuat terhadap perencanaan, pengendalian proses, dan stabilitas kebutuhan selama siklus pengembangan.

2. Kekuatan Waterfall

Kekuatan utama Waterfall salah satunya adalah struktur proses yang jelas dan tingkat prediktabilitas yang relatif tinggi. Perencanaan kebutuhan, ruang lingkup, jadwal, dan sumber daya sejak awal memberikan kerangka kerja yang dapat digunakan untuk mengendalikan pelaksanaan proyek (Mishra & Alzoubi, 2023). Dokumentasi yang sistematis juga mendukung keterlacakan proses, transfer pengetahuan, dan pengendalian proyek, terutama pada organisasi yang memiliki struktur pengembangan kompleks (Hassani & Idrissi, 2022). Kondisi tersebut menjadikan Waterfall relevan untuk proyek dengan kebutuhan yang relatif stabil dan telah dipahami sejak awal pengembangan (Parnell dkk., 2020).

Waterfall juga memiliki relevansi pada proyek yang membutuhkan dokumentasi, verifikasi, validasi, dan kepatuhan terhadap persyaratan regulasi secara ketat (Ribeiro dkk., 2024). Selain itu, struktur tahapan yang terdefinisi dapat membantu pengelolaan proyek berskala besar melalui pembagian aktivitas, estimasi waktu, dan pengendalian sumber daya yang lebih sistematis (Saravanos & Curinga, 2023). Kesesuaian Waterfall juga dipengaruhi oleh karakteristik organisasi. Organisasi dengan struktur hierarkis, prosedur formal, dan pengambilan keputusan yang terpusat cenderung memiliki lingkungan yang lebih sesuai dengan pendekatan pengembangan berbasis perencanaan (Vatan dkk., 2022).

3. Kelemahan Waterfall

Keterbatasan utama Waterfall terletak pada kemampuannya dalam merespons perubahan kebutuhan selama proses pengembangan. Model ini mengandalkan asumsi bahwa kebutuhan dapat didefinisikan secara relatif lengkap pada tahap awal, sehingga perubahan setelah tahap tersebut selesai dapat berdampak pada tahapan berikutnya (Hassani & Idrissi, 2022). Keterbatasan tersebut menjadi lebih signifikan pada lingkungan pengembangan yang memiliki tingkat perubahan kebutuhan dan teknologi yang tinggi. Perubahan yang terjadi setelah suatu tahap diselesaikan dapat memerlukan revisi terhadap dokumentasi atau pekerjaan sebelumnya sehingga meningkatkan waktu dan biaya pengembangan (Pandey & Litoriya, 2021).

Waterfall juga memiliki keterbatasan dalam memberikan umpan balik pengguna secara berkelanjutan karena keterlibatan pengguna umumnya lebih intensif pada tahap awal dan evaluasi hasil dilakukan setelah produk dikembangkan (Dendere dkk., 2021). Kondisi tersebut dapat meningkatkan risiko ketidaksesuaian antara perangkat lunak yang dihasilkan dan kebutuhan aktual pengguna apabila kebutuhan mengalami perubahan selama pengembangan (Diansyah dkk., 2023). Selain itu, proses dokumentasi dan pengendalian yang bersifat formal dapat meningkatkan beban administratif pada proyek yang membutuhkan respons cepat terhadap perubahan (Vukadinović & Fabac, 2022). Oleh karena itu, efektivitas Waterfall sangat dipengaruhi oleh stabilitas kebutuhan, tingkat perubahan lingkungan, kompleksitas proyek, dan tuntutan adaptasi selama siklus pengembangan (Mishra & Alzoubi, 2023).

Metode Pengembangan Agile

1. Karakteristik Agile

Agile berkembang pada awal tahun 2000-an sebagai pendekatan yang merespons keterbatasan model pengembangan perangkat lunak tradisional, dengan prinsip yang dirumuskan melalui *Agile Manifesto* pada tahun 2001 (Mergel dkk., 2021). Prinsip tersebut menekankan interaksi antarmanusia, perangkat lunak yang

dapat digunakan, kolaborasi dengan pelanggan, dan respons terhadap perubahan (Thota & Jung, 2024). Agile bukan merupakan satu metodologi tunggal, melainkan kumpulan nilai dan prinsip yang diimplementasikan melalui berbagai kerangka kerja, seperti Scrum, Extreme Programming (XP), Kanban, dan Scaled Agile Framework (SAFe) (Pandey & Litoriya, 2021). Karakteristik utama Agile adalah pengembangan iteratif dan inkremental, sehingga perangkat lunak dapat dikembangkan, diuji, dan dievaluasi secara bertahap berdasarkan umpan balik yang diperoleh selama proses pengembangan (Diansyah dkk., 2023).

Aktivitas dalam setiap iterasinya mulai dari analisis kebutuhan, perancangan, implementasi, dan pengujian dapat dilakukan secara berulang sehingga perubahan kebutuhan dapat diakomodasi sepanjang siklus pengembangan (Ribeiro dkk., 2024). Pendekatan tersebut memungkinkan tim memperoleh umpan balik lebih awal dan melakukan penyesuaian terhadap produk berdasarkan hasil evaluasi setiap iterasi (Mishra & Alzoubi, 2023). Agile juga menekankan kolaborasi, komunikasi, keterlibatan pelanggan, dan pengambilan keputusan yang adaptif dalam proses pengembangan (Grinberg & Niță, 2022). Scrum menggunakan *Sprint* sebagai periode pengembangan iteratif, XP menekankan praktik teknis seperti *pair programming* dan *test-driven development*, Kanban berfokus pada visualisasi alur kerja dan pembatasan pekerjaan yang sedang berlangsung, sedangkan SAFe mendukung penerapan prinsip Agile pada skala organisasi yang lebih besar (Pavličková dkk., 2024).

2. Kekuatan Agile

Kekuatan utama Agile terletak pada fleksibilitas dan kemampuan beradaptasi terhadap perubahan. Pengembangan secara iteratif memungkinkan tim menyesuaikan prioritas dan produk berdasarkan perubahan kebutuhan pengguna serta kondisi proyek selama proses pengembangan berlangsung (Hassani & Idrissi, 2022). Keterlibatan pemangku kepentingan secara berkelanjutan memungkinkan umpan balik digunakan sebagai dasar penyempurnaan pada iterasi berikutnya sehingga risiko ketidaksesuaian antara produk dan kebutuhan pengguna dapat dikurangi (Mishra & Alzoubi, 2023). Selain itu, pendekatan iteratif memungkinkan perangkat lunak disampaikan secara bertahap sehingga manfaat sistem dapat diperoleh sebelum keseluruhan produk selesai (Thota & Jung, 2024).

Pembagian pengembangan ke dalam iterasi juga mendukung identifikasi dan pengelolaan risiko secara lebih awal karena setiap bagian produk dapat dievaluasi sebelum pengembangan dilanjutkan ke bagian berikutnya (Elkhatib dkk., 2022). Karakteristik tersebut menjadikan Agile relevan untuk proyek yang memiliki tingkat ketidakpastian, kompleksitas, dan perubahan kebutuhan yang relatif tinggi (Vukadinović & Fabac, 2022).

3. Kelemahan Agile

Agile memiliki tingkat fleksibilitas yang tinggi akan tetapi memiliki keterbatasan dalam dokumentasi dan pengelolaan pengetahuan karena penekanannya lebih besar pada interaksi, kolaborasi, dan perangkat lunak yang dapat digunakan (Hassani & Idrissi, 2022). Keterbatasan tersebut dapat menjadi tantangan bagi organisasi besar atau proyek yang membutuhkan dokumentasi formal untuk mendukung transfer pengetahuan, audit, dan kepatuhan terhadap regulasi (Dietze & Kahrens, 2022). Fleksibilitas yang tinggi juga dapat meningkatkan risiko perluasan ruang lingkup apabila perubahan kebutuhan tidak dikelola melalui mekanisme prioritas dan pengendalian yang memadai (Elkhatib dkk., 2022). Selain itu, sifat iteratif Agile dapat menyebabkan estimasi keseluruhan waktu dan biaya proyek menjadi lebih dinamis dibandingkan pendekatan yang menetapkan ruang lingkup secara lebih ketat sejak awal (Mishra & Alzoubi, 2023).

Agile juga menghadapi tantangan pada proyek yang memiliki tuntutan keselamatan, verifikasi, validasi, dan regulasi yang tinggi karena fleksibilitas proses perlu diseimbangkan dengan kebutuhan dokumentasi dan pengendalian formal (Ribeiro dkk., 2024). Keberhasilan penerapannya juga dipengaruhi oleh budaya organisasi karena struktur yang sangat hierarkis dan pengambilan keputusan yang terpusat dapat menghambat penerapan kolaborasi dan desentralisasi yang menjadi karakteristik Agile (Vatan dkk., 2022). Pada organisasi besar dan tim yang tersebar, penerapan Agile dapat menimbulkan tantangan koordinasi sehingga diperlukan mekanisme tambahan untuk mengelola ketergantungan antartim dan menyelaraskan proses pengembangan. Pengembangan kerangka kerja seperti SAFe menunjukkan adanya upaya untuk mengadaptasi prinsip Agile agar dapat diterapkan pada lingkungan pengembangan dengan skala dan kompleksitas yang lebih besar (Pavlíčková dkk., 2024).

Analisis Perbandingan Waterfall dan Agile

Berdasarkan hasil sintesis literatur, Waterfall dan Agile memiliki perbedaan pada struktur proses, pengelolaan kebutuhan, dokumentasi, keterlibatan pemangku kepentingan, pengujian, fleksibilitas, dan pengelolaan risiko. Perbedaan tersebut dirangkum dalam Tabel 1.

Tabel 1. Perbandingan Waterfall dan Agile

No	Dimensi	Waterfall	Agile
1	Struktur	Linear dan berurutan	Iteratif dan inkremental
2	Kebutuhan	Didefinisikan sejak awal	Berkembang selama proyek
3	Dokumentasi	Komprehensif	Relatif lebih ringan
4	Keterlibatan pengguna	Relatif terbatas hingga tahap akhir	Berkelanjutan
5	Pengujian	Umumnya dilakukan setelah implementasi	Terintegrasi sepanjang pengembangan
6	Perubahan	Sulit dan relatif mahal	Diakomodasi secara iteratif
7	Pengelolaan risiko	Lebih formal dan terpisah	Terintegrasi dalam proses
8	Prediktabilitas	Relatif tinggi ketika kebutuhan stabil	Relatif lebih rendah pada awal proyek
9	Fleksibilitas	Rendah	Tinggi
10	Konteks yang sesuai	Kebutuhan stabil dan terdefinisi	Kebutuhan berkembang dan lingkungan dinamis

Perbedaan kedua pendekatan tidak hanya terletak pada urutan aktivitas pengembangan, tetapi juga pada cara metodologi mengelola ketidakpastian, perubahan kebutuhan, keterlibatan pengguna, dan pengendalian proses pengembangan (Hassani & Idrissi, 2022). Pemilihan metodologi pengembangan perangkat lunak perlu mempertimbangkan karakteristik proyek dan lingkungan organisasi karena tidak terdapat satu pendekatan yang sesuai untuk seluruh kondisi pengembangan (Mitchell & Mitchell, 2024). Waterfall cenderung sesuai untuk proyek dengan kebutuhan yang relatif stabil, ruang lingkup yang telah ditentukan, serta tuntutan dokumentasi dan kepatuhan yang tinggi (Thota & Jung, 2024). Agile lebih sesuai pada lingkungan pengembangan dengan kebutuhan yang dinamis, tingkat ketidakpastian yang tinggi, dan

kebutuhan terhadap umpan balik pengguna secara berkelanjutan (Vukadinović & Fabac, 2022). Karakteristik tersebut memungkinkan proses pengembangan disesuaikan secara bertahap berdasarkan perubahan kebutuhan dan hasil evaluasi selama iterasi (Hassani & Idrissi, 2022).

Budaya organisasi merupakan faktor yang memengaruhi pemilihan dan penerapan metodologi pengembangan perangkat lunak selain karakteristik teknis. Organisasi dengan struktur hierarkis dan aturan yang kuat cenderung lebih mendukung pendekatan yang terstruktur, sedangkan lingkungan organisasi yang mendorong kolaborasi dan adaptabilitas dapat mendukung penerapan Agile (Vatan dkk., 2022). Pemilihan metodologi juga dipengaruhi oleh cara manajer proyek menyesuaikan proses pengembangan dengan karakteristik dan budaya organisasi (Piwowar-Sulej, 2021). Oleh karena itu, faktor seperti kesiapan organisasi, kompetensi tim, pola pengambilan keputusan, keterlibatan pengguna, kompleksitas proyek, serta kemampuan menerima perubahan perlu dipertimbangkan dalam menentukan pendekatan pengembangan yang digunakan.

Pendekatan Hybrid Waterfall-Agile

Keterbatasan Waterfall dalam merespons perubahan dan tantangan penerapan Agile pada proyek yang membutuhkan perencanaan serta pengendalian formal mendorong berkembangnya pendekatan *hybrid* yang mengintegrasikan elemen dari kedua paradigma pengembangan. Pendekatan *hybrid* dirancang untuk mempertahankan struktur, perencanaan, dan pengendalian proses yang menjadi karakteristik Waterfall, sekaligus mengadopsi fleksibilitas dan pengembangan iteratif dari Agile (Reiff & Schlegel, 2022). Pendekatan tersebut didasarkan pada pertimbangan bahwa karakteristik proyek tidak selalu dapat dikategorikan secara sederhana sebagai stabil atau dinamis. Beberapa proyek memiliki kebutuhan yang relatif terdefinisi pada tahap awal sehingga memerlukan perencanaan dan dokumentasi formal, tetapi pada tahap pengembangan tetap menghadapi perubahan kebutuhan yang memerlukan mekanisme adaptasi. Oleh karena itu, pendekatan *hybrid* memberikan ruang bagi organisasi untuk menyesuaikan proses pengembangan berdasarkan karakteristik dan kebutuhan spesifik proyek (Hassani & Idrissi, 2022).

Penerapan pola *hybrid* salah satunya adalah penggunaan pendekatan Waterfall pada tahap perencanaan dan pengelolaan kebutuhan, kemudian menerapkan praktik Agile pada tahap pengembangan dan penyampaian produk. Pola tersebut sering dikaitkan dengan istilah Water-Scrum-Fall, yaitu pendekatan yang mempertahankan proses perencanaan dan pengendalian tradisional serta mengintegrasikan praktik Scrum dalam aktivitas pengembangan (Piwowar-Sulej, 2021). Pola *hybrid* lainnya adalah Waterfall-Agile Method (WAM) yang mengintegrasikan karakteristik Waterfall dan Agile dalam tahapan pengembangan perangkat lunak. Dalam penerapannya, perencanaan dapat dilakukan menggunakan pendekatan Waterfall, sedangkan aktivitas pengembangan dilaksanakan secara iteratif menggunakan Scrum, dengan pengujian dan penerapan disesuaikan dengan karakteristik serta kebutuhan proyek (Yahya & Maidin, 2023). Pada lingkungan pengembangan berskala besar, integrasi praktik Agile juga dapat dilakukan melalui kerangka kerja yang mendukung koordinasi antartim, seperti Scaled Agile Framework (SAFe). Sementara itu, pada proyek yang memiliki tuntutan verifikasi dan validasi yang ketat, prinsip Agile dapat dikombinasikan dengan pendekatan berbasis *V-Model* untuk mempertahankan pengendalian terhadap proses pengujian dan validasi (Pavličková dkk., 2024).

Pendekatan *hybrid* memungkinkan organisasi menyesuaikan tingkat perencanaan, dokumentasi, iterasi, dan keterlibatan pemangku kepentingan sesuai dengan kebutuhan proyek. Integrasi tersebut dapat memberikan fleksibilitas dalam menghadapi perubahan sekaligus mempertahankan mekanisme

pengendalian yang diperlukan untuk memenuhi persyaratan organisasi maupun regulasi (Reiff & Schlegel, 2022). Selain itu, pendekatan *hybrid* dapat digunakan ketika karakteristik proyek membutuhkan kombinasi antara prediktabilitas pada bagian tertentu dan kemampuan adaptasi pada bagian lainnya. Dengan pembagian penerapan metodologi berdasarkan tahapan atau karakteristik aktivitas, organisasi dapat menyesuaikan proses pengembangan tanpa harus menerapkan satu pendekatan secara menyeluruh pada seluruh siklus hidup perangkat lunak (Mishra & Alzoubi, 2023). Namun, penerapan *hybrid* juga memiliki tantangan, terutama karena tidak terdapat satu pola *hybrid* yang dapat diterapkan secara universal pada seluruh proyek. Variasi pola penerapan menyebabkan organisasi perlu menentukan kombinasi praktik, batas tanggung jawab, mekanisme koordinasi, serta kriteria pengendalian perubahan secara jelas (Prenner dkk., 2021). Selain itu, integrasi Waterfall dan Agile dapat menimbulkan ketidaksesuaian antara kebutuhan perencanaan dan pengendalian yang bersifat prediktif dengan kebutuhan adaptasi yang bersifat iteratif. Oleh karena itu, penerapan *hybrid* memerlukan kejelasan proses, koordinasi antartim, kompetensi sumber daya manusia, serta kesesuaian dengan struktur dan budaya organisasi (Piwowar-Sulej, 2021).

Pembahasan dan Sintesis

Hasil kajian menunjukkan bahwa perbandingan Waterfall dan Agile tidak lagi memadai apabila dipahami semata-mata sebagai pertentangan antara metodologi tradisional dan modern. Literatur menunjukkan adanya pergeseran perspektif menuju pemilihan metodologi yang lebih kontekstual, yaitu dengan mempertimbangkan karakteristik proyek, lingkungan pengembangan, serta kebutuhan organisasi (Reiff & Schlegel, 2022). Waterfall cenderung sesuai pada proyek dengan kebutuhan yang relatif stabil, ruang lingkup yang terdefinisi, serta tuntutan terhadap dokumentasi, keterlacakan, dan kepatuhan regulasi. Sebaliknya, Agile memiliki karakteristik yang mendukung proyek dengan tingkat ketidakpastian dan perubahan kebutuhan yang tinggi melalui pengembangan iteratif serta umpan balik berkelanjutan (Hassani & Idrissi, 2022).

Perbedaan tersebut menunjukkan bahwa karakteristik yang menjadi keterbatasan suatu metodologi dalam konteks tertentu dapat memberikan manfaat pada konteks yang berbeda. Dokumentasi yang ekstensif pada Waterfall, misalnya, dapat mendukung audit, keterlacakan, dan kepatuhan pada proyek yang memiliki persyaratan formal, tetapi dapat meningkatkan beban proses pada proyek yang membutuhkan perubahan secara cepat. Sebaliknya, fleksibilitas Agile memungkinkan penyesuaian kebutuhan secara iteratif, tetapi dapat menimbulkan tantangan ketika proyek memerlukan prediktabilitas ruang lingkup, anggaran, dokumentasi formal, dan mekanisme verifikasi yang ketat (Mishra & Alzoubi, 2023). Selain karakteristik proyek, budaya organisasi merupakan faktor penting dalam pemilihan dan penerapan metodologi pengembangan perangkat lunak. Penerapan Agile memerlukan kolaborasi, komunikasi, keterlibatan pemangku kepentingan, serta pola pengambilan keputusan yang mendukung adaptasi terhadap perubahan. Apabila struktur dan budaya organisasi belum mendukung karakteristik tersebut, penerapan Agile dapat menghadapi hambatan meskipun kondisi proyek secara teknis memungkinkan penggunaan pendekatan iteratif (Vatan dkk., 2022).

Kajian juga mengidentifikasi penggunaan pendekatan *hybrid* sebagai bentuk adaptasi terhadap kebutuhan proyek yang memiliki karakteristik berbeda pada tahapan pengembangannya. Pendekatan ini memungkinkan organisasi mengintegrasikan mekanisme perencanaan dan pengendalian dari Waterfall dengan praktik iteratif dan adaptif dari Agile sesuai kebutuhan proyek (Reiff & Schlegel, 2022). Dari perspektif pengembangan perangkat lunak, pendekatan *hybrid* tidak berarti bahwa seluruh tahapan harus

menggabungkan Waterfall dan Agile secara bersamaan. Integrasi dapat dilakukan berdasarkan tahapan, komponen, tingkat risiko, atau karakteristik kebutuhan yang berbeda dalam satu proyek sehingga mekanisme pengembangan dapat disesuaikan dengan kondisi masing-masing bagian proyek (Prenner dkk., 2021). Berdasarkan sintesis tersebut, pemilihan metodologi pengembangan perangkat lunak perlu mempertimbangkan karakteristik proyek, tingkat ketidakpastian, stabilitas kebutuhan, tuntutan regulasi, budaya organisasi, kompetensi tim, serta tingkat keterlibatan pemangku kepentingan. Faktor-faktor tersebut menentukan sejauh mana suatu pendekatan dapat diterapkan dan disesuaikan selama siklus hidup pengembangan perangkat lunak (Mitchell & Mitchell, 2024).

Secara konseptual, proyek dengan kebutuhan relatif stabil, ruang lingkup yang jelas, serta tuntutan dokumentasi dan regulasi yang tinggi memiliki karakteristik yang mendukung penerapan Waterfall. Proyek dengan kebutuhan yang berkembang, ketidakpastian tinggi, dan kebutuhan terhadap umpan balik pengguna secara cepat memiliki karakteristik yang mendukung Agile. Sementara itu, proyek dengan karakteristik yang berbeda pada bagian atau tahapan tertentu dapat mempertimbangkan pendekatan *hybrid* sebagai mekanisme penyesuaian proses pengembangan. Dengan demikian, adaptabilitas metodologi tidak hanya ditentukan oleh karakteristik internal Waterfall atau Agile, tetapi juga oleh kesesuaian antara metodologi, karakteristik proyek, dan lingkungan organisasi. Temuan ini memperkuat bahwa pemilihan metodologi merupakan keputusan kontekstual yang perlu mempertimbangkan kebutuhan pengembangan dan kemampuan organisasi dalam mengelola perubahan, bukan sekadar membandingkan keunggulan dan keterbatasan masing-masing pendekatan.

KESIMPULAN

Kajian literatur menunjukkan bahwa Waterfall dan Agile memiliki karakteristik, kekuatan, dan keterbatasan yang berbeda dalam pengembangan perangkat lunak dan sistem informasi. Waterfall menekankan perencanaan terstruktur, dokumentasi, prediktabilitas, dan pengendalian proses sehingga karakteristiknya relevan untuk proyek dengan kebutuhan yang relatif stabil, ruang lingkup yang terdefinisi, serta tuntutan dokumentasi dan kepatuhan yang tinggi. Agile menerapkan pengembangan secara iteratif dan inkremental dengan menekankan fleksibilitas, kolaborasi, keterlibatan pemangku kepentingan, serta umpan balik berkelanjutan. Karakteristik tersebut mendukung pengembangan pada proyek dengan kebutuhan yang berkembang, tingkat ketidakpastian yang tinggi, dan lingkungan yang mengalami perubahan secara dinamis. Kedua pendekatan juga memiliki keterbatasan yang perlu dipertimbangkan dalam penerapannya. Waterfall memiliki tantangan dalam mengakomodasi perubahan kebutuhan setelah tahapan pengembangan ditetapkan, sedangkan Agile menghadapi tantangan dalam pengelolaan dokumentasi, pengendalian ruang lingkup, prediktabilitas waktu dan biaya, koordinasi tim, serta penerapan pada proyek yang memiliki tuntutan verifikasi, validasi, dan regulasi yang ketat.

Temuan utama kajian menunjukkan bahwa pemilihan metodologi pengembangan perangkat lunak bersifat kontekstual dan tidak dapat ditentukan hanya berdasarkan karakteristik metodologi secara umum. Pemilihan perlu mempertimbangkan stabilitas kebutuhan, kompleksitas proyek, tingkat ketidakpastian, tuntutan regulasi, budaya organisasi, kompetensi tim, keterlibatan pemangku kepentingan, serta kemampuan organisasi dalam mengelola perubahan. Penggunaan pendekatan *hybrid* menunjukkan adanya kebutuhan untuk menyesuaikan proses pengembangan dengan karakteristik proyek yang beragam. Pendekatan tersebut memungkinkan integrasi praktik perencanaan dan pengendalian dengan mekanisme

iteratif dan adaptif pada bagian atau tahapan tertentu sesuai kebutuhan proyek.

DAFTAR PUSTAKA

- Beasley, R., Woodcock, H., & Eastwood, P. (2022). Mindful Maturation Matters: The V-Model as a Passport to Engagement. *INCOSE International Symposium*, 32(1), 277–292. <https://doi.org/10.1002/iis2.12931>
- Dendere, R., Janda, M., & Sullivan, C. (2021). Are we Doing it Right? We Need to Evaluate the Current Approaches for Implementation of Digital Health Systems. *Australian Health Review*, 45(6), 778–781. <https://doi.org/10.1071/AH20289>
- Diansyah, A. F., Rahman, M. R., Handayani, R., Cahyo, D. D. N., & Utami, E. (2023). Comparative Analysis of Software Development Lifecycle Methods in Software Development: A Systematic Literature Review. *International Journal of Advances in Data and Information Systems*, 4(2), 97–106. <https://doi.org/10.25008/ijadis.v4i2.1295>
- Dietze, M., & Kahrens, M. (2022). Knowledge Activities Applied | Towards a Holistic Knowledge Management Approach in the Software Industry. *VINE Journal of Information and Knowledge Management Systems*, 54(2), 398–423. <https://doi.org/10.1108/VJIKMS-09-2021-0175>
- Elkhatib, M., Hosani, A. Al, Hosani, I. Al, & Albuflasa, K. (2022). Agile Project Management and Project Risks Improvements: Pros and Cons. *Modern Economy*, 13, 1157–1176. <https://doi.org/10.4236/me.2022.139061>
- Grinberg, R., & Niță, V. (2022). Delays of Construction Projects-The Traditional Way or the Hybrid Way? *Review of Economic and Business Studies*, 15(1), 87–98. <https://doi.org/10.47743/rebs-2022-1-0005>
- Hassani, R., & Idrissi, Y. E. B. El. (2022). IT Project Management Models in an Era of Digital Transformation: A Study by Practice. *Advances in Science, Technology and Engineering Systems Journal*, 7(2), 53–62. <https://doi.org/10.25046/aj070205>
- Mergel, I., Ganapati, S., & Whitford, A. B. (2021). Agile: A New Way of Governing. *Public Administration Review*, 81(1), 161–165. <https://doi.org/10.1111/puar.13202>
- Mishra, A., & Alzoubi, Y. I. (2023). Structured Software Development Versus Agile Software Development: A Comparative Analysis. *International Journal of System Assurance Engineering and Management*, 14(4), 1504–1522. <https://doi.org/10.1007/s13198-023-01958-5>
- Mitchell, D., & Mitchell, S. (2024). Don't Go Chasing Waterfalls? How Complexity Demands More Agile Project Management Approaches to Municipal Strategy Implementation. *State and Local Government Review*, 56(2), 137–156. <https://doi.org/10.1177/0160323X231215056>
- Pandey, P., & Litoriya, R. (2021). Software Process Selection System Based on Multicriteria Decision Making. *Journal of Software: Evolution and Process*, 33(2). <https://doi.org/https://doi.org/10.1002/smr.2305>
- Parnell, B., Stone, M., & Aravopoulou, E. (2020). Controlling Superprojects-Information Management Requirements. *The Bottom Line*, 33(1), 116–131. <https://doi.org/10.1108/BL-11-2019-0126>
- Pavličková, M., Mojžišová, A., Bodíková, Z., Szeplaki, R., & Laciak, M. (2024). Integration and Implementation of Scaled Agile Framework and V-Model in the Healthcare Sector Organization. *Electronics*, 13(11), 2051. <https://doi.org/10.3390/electronics13112051>
- Piowar-Sulej, K. (2021). Organizational Culture and Project Management Methodology: Research in the Financial Industry. *International Journal of Managing Projects in Business*, 14(6), 1270–1289. <https://doi.org/10.1108/IJMPB-08-2020-0252>
- Prenner, N., Unger-Windeler, C., & Schneider, K. (2021). Goals and Challenges in Hybrid Software Development Approaches. *Journal of Software: Evolution and Process*, 33(11). <https://doi.org/https://doi.org/10.1002/smr.2382>

- Reiff, J., & Schlegel, D. (2022). Hybrid Project Management – A Systematic Literature Review. *International Journal of Information Systems and Project Management*, 10(2), 45–63. <https://doi.org/10.12821/ijispm100203>
- Ribeiro, J. E. F., Silva, J. G., & Aguiar, A. (2024). Weaving Agility in Safety-Critical Software Development for Aerospace: From Concerns to Opportunities. *IEEE Access*, 12, 52778–52802. <https://doi.org/10.1109/ACCESS.2024.3387730>
- Saravanos, A., & Curinga, M. X. (2023). Simulating the Software Development Lifecycle: The Waterfall Model. *Applied System Innovation*, 6(6), 108. <https://doi.org/10.3390/asi6060108>
- Thota, A. K., & Jung, R. (2024). Accelerating Neurotechnology Development Using an Agile Methodology. *Frontiers in Neuroscience*, 18. <https://doi.org/10.3389/fnins.2024.1328540>
- Vatan, E., Ardali, G. A. R., & Shahin, A. (2022). Selecting Information Systems Development Models Based on Organizational Culture: An Integrated Approach of DEMATEL and ANP. *VINE Journal of Information and Knowledge Management Systems*, 54(3), 531–560. <https://doi.org/10.1108/VJIKMS-08-2021-0164>
- Vukadinović, D., & Fabac, R. (2022). Project Management in Digital Disruption: Emergence of Digital Project Management Office. *Entrenova - Enterprise Research Innovation*, 8(1), 310–328. <https://doi.org/10.54820/entrenova-2022-0027>
- Yahya, N., & Maidin, S. (2023). Hybrid Agile Development Phases: The Practice in Software Projects as Performed by Software Engineering Team. *Indonesian Journal of Electrical Engineering and Computer Science*, 29(3), 1738. <https://doi.org/10.11591/ijeecs.v29.i3.pp1738-1749>