

MENGOPTIMALKAN KOMUNIKASI DALAM TIM PENGEMBANGAN PERANGKAT LUNAK MELALUI PENDEKATAN AGILE DENGAN SCRUM: LITERATURE REVIEW

Willy Azrieel, Natalio Valentino

Program Studi Teknik Informatika S1, Fakultas Teknik, Universitas Palangka Raya
Jl. Yos Sudarso, Palangka, Kec. Jekan Raya, Kota Palangka Raya, Kalimantan Tengah 74874, Indonesia
willyazrieelbelajar@gmail.com

ABSTRAK

Komunikasi yang efektif dalam tim pengembangan perangkat lunak merupakan faktor kunci kesuksesan proyek. Scrum adalah kerangka kerja yang mendukung kolaborasi antar anggota tim untuk menghasilkan perangkat lunak dengan iterasi yang cepat dan berkelanjutan. Permasalahan utama yang dihadapi adalah rendahnya efektivitas komunikasi dalam tim yang dapat mengakibatkan kesalahpahaman, keterlambatan proyek, dan penurunan kualitas produk akhir. Tujuan dari penelitian ini adalah untuk mengidentifikasi bagaimana pendekatan Agile dengan Scrum dapat mengatasi hambatan komunikasi dan meningkatkan produktivitas serta kualitas produk. Metode yang digunakan dalam penelitian ini adalah literature review, di mana data dikumpulkan dari berbagai jurnal terakreditasi dan dianalisis untuk mengidentifikasi tema-tema utama terkait implementasi Scrum. Dengan menggunakan metode Scrum, tim dapat meningkatkan transparansi, inspeksi, dan adaptasi, yang semuanya berkontribusi pada komunikasi yang lebih baik dan hasil proyek yang lebih sukses. Hasil penelitian menunjukkan bahwa Scrum dapat meningkatkan transparansi, mengurangi hambatan komunikasi, meningkatkan produktivitas tim, dan memperbaiki kualitas produk akhir.

Kata kunci: Agile, Scrum, komunikasi tim, pengembangan perangkat lunak, inspeksi, adaptasi

1. PENDAHULUAN

Metode Scrum telah menjadi standar dalam pengembangan perangkat lunak modern karena kemampuannya untuk meningkatkan kolaborasi dan komunikasi antar anggota tim. Di lingkungan kerja yang kompleks dan dinamis, komunikasi yang efektif adalah kunci untuk menyelesaikan proyek dengan sukses. Komunikasi yang buruk dapat menyebabkan kesalahpahaman, keterlambatan proyek, dan penurunan kualitas produk akhir.

Dalam konteks pengembangan perangkat lunak, tantangan komunikasi sering kali muncul karena berbagai alasan, termasuk perbedaan dalam pemahaman teknis, jarak geografis, dan kurangnya keterlibatan tim secara menyeluruh. Untuk mengatasi masalah ini, berbagai metodologi telah dikembangkan, salah satunya adalah Scrum. Scrum adalah salah satu metode agile yang paling banyak digunakan, yang dirancang untuk mengatasi kompleksitas proyek dengan membagi pekerjaan menjadi iterasi-iterasi yang lebih kecil dan dapat dikelola.

Scrum menekankan pentingnya tim yang mandiri dan kolaboratif, di mana setiap anggota tim memiliki tanggung jawab yang jelas dan berkontribusi terhadap tujuan bersama. Elemen-elemen inti dalam Scrum termasuk peran (Product Owner, Scrum Master, dan Development Team), acara (Daily Stand-up, Sprint Planning, Sprint Review, dan Sprint Retrospective), dan artefak (Product Backlog, Sprint Backlog, dan Increment). Semua elemen ini dirancang untuk meningkatkan transparansi, inspeksi, dan adaptasi dalam tim.

Untuk mengoptimalkan komunikasi dalam tim pengembangan perangkat lunak menggunakan metodologi Scrum, penting untuk fokus pada aspek-aspek kunci yang disorot dalam literatur. Scrum, sebagai model agile yang banyak dipraktikkan, menekankan pada organisasi mandiri yang efektif, komunikasi, kolaborasi, dan peningkatan berkelanjutan [1].

Scrum membantu mencapai kualitas perangkat lunak dengan mempromosikan kolaborasi, keamanan psikologis, akuntabilitas, transparansi, pengembangan iteratif, inspeksi formal, dan adaptasi [2]. Selain itu, kerangka kerja Scrum mendorong komunikasi reguler dengan pemilik produk melalui iterasi berkelanjutan, meningkatkan keterlibatan pelanggan dalam proses pengembangan [3].

Metodologi ini memerlukan serangkaian kompetensi yang berbeda, termasuk pengetahuan, keterampilan, dan sikap, agar anggota tim dapat bekerja sama secara efektif [4].

Dalam dunia pengembangan perangkat lunak, tantangan komunikasi sering terjadi dan dapat berdampak signifikan pada keberhasilan proyek. Metodologi Agile, seperti Scrum, telah banyak diadopsi untuk mengatasi tantangan ini dengan menekankan kolaborasi, adaptabilitas, dan proses pengembangan iteratif [5]. Peralihan menuju pengembangan agile yang terdistribusi semakin memperumit dinamika komunikasi, memerlukan peninjauan kembali praktik-praktik tradisional untuk memastikan koordinasi yang lancar di antara tim yang tersebar secara geografis [6].

2. TINJAUAN PUSTAKA

2.1. Metode Agile

Metode Agile dalam pengembangan perangkat lunak menekankan fleksibilitas, kolaborasi, dan iterasi cepat. Agile muncul sebagai respons terhadap keterbatasan pendekatan tradisional seperti Waterfall, yang sering kali kurang responsif terhadap perubahan kebutuhan dan lingkungan proyek yang dinamis. Agile bertujuan untuk meningkatkan kemampuan tim dalam menanggapi perubahan dan meningkatkan komunikasi antar anggota tim serta dengan pemangku kepentingan [3].

2.2. Implementasi Agile dengan Scrum dan Komunikasi dalam Tim

Scrum adalah salah satu kerangka kerja yang paling banyak digunakan dalam implementasi Agile. Scrum berfokus pada pengelolaan proyek melalui iterasi yang disebut sprint, yang biasanya berlangsung selama dua hingga empat minggu. Setiap sprint dimulai dengan perencanaan sprint dan diakhiri dengan review dan retrospektif sprint. Scrum memiliki tiga peran utama yaitu Product Owner, Scrum Master, dan Development Team, serta menggunakan artefak seperti Product Backlog, Sprint Backlog, dan Increment untuk mendukung transparansi dan inspeksi [1].

Dalam konteks komunikasi tim, Scrum memiliki beberapa mekanisme yang dirancang untuk meningkatkan transparansi dan kolaborasi. Daily Stand-up memungkinkan anggota tim untuk memberikan pembaruan harian dan mengidentifikasi hambatan yang menghalangi mereka. Sprint Planning membantu tim untuk menetapkan tujuan sprint dan menentukan pekerjaan yang akan dilakukan. Sprint Review memberikan kesempatan untuk mendemonstrasikan hasil kerja kepada pemangku kepentingan dan mendapatkan umpan balik, sementara Sprint Retrospective fokus pada refleksi dan perbaikan proses tim secara berkelanjutan [1] [2].

2.3. Peran dalam Scrum

Scrum memiliki tiga peran utama: Product Owner, Scrum Master, dan Development Team. Setiap peran memiliki tanggung jawab yang spesifik yang mendukung komunikasi yang efektif. Product Owner bertanggung jawab untuk mengelola backlog produk dan memastikan tim memahami prioritas proyek. Scrum Master bertugas menghilangkan hambatan yang menghalangi kinerja tim. Development Team bertanggung jawab untuk menghasilkan increment yang berfungsi setiap sprint [2].

2.4. Acara dalam Scrum

Acara dalam Scrum seperti Daily Stand-up, Sprint Planning, Sprint Review, dan Sprint Retrospective dirancang untuk memastikan

komunikasi yang berkelanjutan dan inspeksi serta adaptasi yang konstan. Daily Stand-up, misalnya, memungkinkan anggota tim untuk memberikan pembaruan harian dan mengidentifikasi hambatan yang menghalangi mereka [3].

3. METODE PENELITIAN

Penelitian ini menggunakan metode literature review untuk mengkaji dampak implementasi metode Scrum terhadap komunikasi dalam tim pengembangan perangkat lunak. Literature review ini bertujuan untuk mengumpulkan dan menganalisis data dari berbagai penelitian yang telah dilakukan mengenai topik ini. Proses literature review melibatkan beberapa tahap, yaitu:

3.1. Identifikasi Sumber

Pencarian literatur dilakukan menggunakan kata kunci seperti "Scrum", "komunikasi tim", "pengembangan perangkat lunak", "inspeksi", dan "adaptasi". Jurnal-jurnal yang diidentifikasi sebagai relevan kemudian dievaluasi berdasarkan topik, kualitas penelitian, dan tanggal publikasi.

3.2. Seleksi dan Evaluasi

Sumber-sumber yang digunakan adalah jurnal-jurnal yang dipublikasikan jurnal internasional atau jurnal nasional terakreditasi minimal Sinta 4 dari tahun 2018 hingga 2024, menggunakan bantuan software Harzing's Publish or Perish 8 untuk mencari jurnal dan sumber lainnya seperti Google Scholars, ScienceDirect, dan lain-lain. Kriteria seleksi mencakup relevansi topik dengan penelitian ini, serta kualitas metodologi yang digunakan dalam penelitian tersebut.

3.3. Analisis Data

Data dari jurnal yang terpilih dianalisis untuk mengidentifikasi tema-tema utama dan tren yang muncul dalam penelitian terkait. Analisis ini mencakup bagaimana metode Scrum mempengaruhi komunikasi dalam tim, hambatan-hambatan yang ada, serta manfaat yang diperoleh dari implementasi Scrum.

3.4. Sintesis Hasil

Analisis kemudian disintesis kan untuk memberikan gambaran yang komprehensif tentang dampak pendekatan agile dengan Scrum terhadap komunikasi dalam tim pengembangan perangkat lunak. Fokus utama adalah pada peningkatan transparansi, pengurangan hambatan komunikasi, peningkatan produktivitas, peningkatan kualitas produk, peningkatan kolaborasi tim, dan adaptabilitas terhadap perubahan.

4. HASIL DAN PEMBAHASAN

Tabel 1. Tinjauan Komprehensif

No	Judul	Tinjauan Komprehensif
1	More than Code: Contributions in Scrum Software Engineering Teams [1]	Menyoroti pentingnya kontribusi tim dalam proyek Scrum yang mencakup aspek teknis dan non-teknis, serta model yang meningkatkan visibilitas tugas dan retrospeksi tim.
2	How Scrum adds value to achieving software quality? [2]	Menjelaskan bagaimana Scrum meningkatkan kualitas perangkat lunak melalui kolaborasi, keamanan psikologis, akuntabilitas, transparansi, dan inspeksi formal.
3	Hybrid agile development phases: The practice in software projects as performed by software engineering team [3]	Meneliti penggunaan model hybrid antara Scrum dan Waterfall untuk memanfaatkan keunggulan masing-masing dan menyesuaikan nya dengan kebutuhan proyek.
4	Scrum team competence based on knowledge, skills, attitude in global software development [4]	Mengembangkan daftar kompetensi tim Scrum berbasis KSA (Knowledge, Skills, Attitude) dalam pengembangan perangkat lunak global.
5	LITERATURE REVIEW: KOMUNIKASI SEBAGAI FAKTOR PENTING DALAM PENGEMBANGAN PERANGKAT LUNAK [7]	Menekankan pentingnya aspek komunikasi dalam pemenuhan kebutuhan rekayasa perangkat lunak yang mempengaruhi keberhasilan proyek.
6	Pengunaan Metode Scrum Dalam Pengembangan Perangkat Lunak: Literature Review [8]	Mengkaji penggunaan metode Scrum dalam pengembangan perangkat lunak dan cara mengatasi masalah terkait proses Scrum yang kompleks.
7	Coordinating Knowledge Work in Multiteam Programs: Findings From a Large-Scale Agile Development Program [9]	Menggambarkan koordinasi kerja pengetahuan dalam program pengembangan agile skala besar dengan 12 tim dan bagaimana praktik koordinasi berubah seiring waktu.
8	Implementing Agile Scrum Methodology in The Development of SICITRA Mobile Application [10]	Menunjukkan bagaimana Scrum yang terdokumentasi dengan baik membantu melacak kemajuan dan mengevaluasi implementasi Scrum dalam pengembangan aplikasi mobile.
9	A framework for risk management in Scrum development process [11]	Mengusulkan kerangka kerja untuk mengintegrasikan manajemen risiko dalam proyek pengembangan Scrum untuk meningkatkan tingkat keberhasilan proyek.
10	Towards Scrum Based Agile Framework for Global Software Development Teams [5]	Mengembangkan solusi Scrum berbasis agile yang disesuaikan untuk tim terdistribusi untuk meningkatkan komunikasi dan koordinasi.
11	An Empirical Investigation of Geographically Distributed Agile Development: The Agile Enterprise Architecture is a Communication Enabler [6]	Meneliti bagaimana arsitektur perusahaan agile dapat meningkatkan komunikasi dan kinerja dalam pengembangan perangkat lunak terdistribusi.
12	Critical Success Factors of Agile Software Development - A Systematic Literature Review [12]	Mengidentifikasi faktor-faktor kunci keberhasilan dalam pengembangan perangkat lunak agile, termasuk kemampuan tim, keterlibatan pelanggan, dan keselamatan psikologis.
13	Agile Processes in Software Engineering and Extreme Programming [13]	Menyediakan panduan komprehensif tentang proses agile dan XP (Extreme Programming) dalam rekayasa perangkat lunak, menekankan pentingnya adaptabilitas dan efisiensi.
14	Enterprise architecture contribution in distributed agile software development [14]	Mengkaji kontribusi arsitektur perusahaan agile dalam meningkatkan komunikasi, kualitas, dan fungsionalitas pengembangan perangkat lunak terdistribusi.
15	The Role of Communication and Meta-communication in Software Engineering with Relation to Human Errors [15]	Menyelidiki peran komunikasi dan meta-komunikasi dalam rekayasa perangkat lunak untuk mengurangi kesalahan manusia dan meningkatkan efektivitas proyek.
16	Investigating the role of Product Owner in Scrum teams: Differentiation between organisational and individual impacts and opportunities [16]	Menganalisis peran Product Owner dalam tim Scrum dan bagaimana peran ini dipengaruhi oleh konteks organisasi dan individu.
17	Agile software development projects–Unveiling the human-related critical success factors [17]	Mengungkap faktor-faktor kunci manusia yang mempengaruhi keberhasilan proyek pengembangan perangkat lunak agile, termasuk kemampuan tim dan keterlibatan pelanggan.

4.1. Peningkatan Transparansi

Implementasi Scrum dalam tim pengembangan perangkat lunak meningkatkan transparansi antar anggota tim. Setiap anggota tim dapat melihat perkembangan proyek melalui artefak Scrum seperti Product Backlog, Sprint Backlog, dan Burndown

Chart. Transparansi ini memungkinkan semua anggota tim untuk memahami status proyek saat ini, tugas yang sedang dikerjakan, dan masalah yang mungkin dihadapi. Transparansi ini sangat penting untuk mengurangi risiko miskomunikasi dan memastikan bahwa semua orang memiliki pemahaman yang sama

tentang proyek. [2] menunjukkan bahwa transparansi dalam Scrum membantu mempromosikan akuntabilitas dan kolaborasi yang lebih baik dalam tim, yang esensial untuk mencapai kualitas perangkat lunak yang lebih tinggi. Selain itu, transparansi ini juga memungkinkan manajer proyek untuk membuat keputusan yang lebih baik berdasarkan data yang akurat dan terkini.

Transparansi juga memfasilitasi identifikasi dan penyelesaian masalah secara lebih cepat. Dalam konteks pengembangan perangkat lunak yang kompleks, kemampuan untuk segera mengetahui di mana letak hambatan atau bottleneck adalah kunci untuk menjaga alur kerja yang efisien. Selain itu, dengan menggunakan alat dan papan visual, anggota tim dapat lebih mudah berbagi informasi dan pemahaman yang sama mengenai prioritas dan tujuan proyek. Menurut penelitian oleh [9], koordinasi yang efektif dalam tim multi-fungsi sangat bergantung pada tingkat transparansi yang tinggi untuk menghindari miskomunikasi dan meningkatkan efisiensi kerja. Pendekatan agile dalam peningkatan transparansi ini adalah penggunaan artefak Scrum yang merupakan praktik Agile untuk memastikan keterbukaan informasi dan kolaborasi yang lebih baik dalam tim.

4.2. Pengurangan Hambatan Komunikasi

Salah satu keuntungan utama dari Scrum adalah pengurangan hambatan komunikasi. Scrum Master memiliki peran penting dalam mengidentifikasi dan menghilangkan hambatan yang mengganggu produktivitas tim. Hambatan ini bisa berupa masalah teknis, konflik antar anggota tim, atau hambatan eksternal seperti keterlambatan dari pihak ketiga. Daily Stand-up meetings adalah elemen kunci dalam Scrum yang memungkinkan tim untuk mendiskusikan hambatan yang mereka hadapi setiap hari. Penelitian oleh [5] menekankan bahwa framework Scrum yang disesuaikan dapat mengatasi tantangan komunikasi dalam tim yang terdistribusi secara geografis, memperkuat koordinasi dan kolaborasi. Dalam tim yang terdistribusi, alat komunikasi digital seperti video conference, chat, dan alat kolaborasi online sangat penting untuk menjaga komunikasi tetap lancar dan efisien.

Selain itu, adanya retrospektif sprint secara rutin memberikan kesempatan bagi tim untuk mendiskusikan apa yang berjalan dengan baik dan apa yang perlu ditingkatkan. Ini tidak hanya membantu dalam menghilangkan hambatan saat ini tetapi juga mencegah munculnya hambatan serupa di masa depan. Komunikasi yang efektif juga diperkuat melalui penggunaan alat kolaborasi digital yang memungkinkan komunikasi sinkron dan asinkron antara anggota tim. [4] menekankan pentingnya kompetensi komunikasi dalam tim global untuk mengatasi tantangan yang muncul dalam lingkungan kerja yang terdistribusi. Pendekatan agile dalam pengurangan hambatan komunikasi ini adalah peran Scrum Master dan pertemuan harian (Daily Stand-up)

yang merupakan praktik Agile untuk memastikan masalah dapat diidentifikasi dan diselesaikan dengan cepat.

4.3. Peningkatan Produktivitas

Tim yang menggunakan Scrum menunjukkan peningkatan produktivitas karena adanya komunikasi yang lebih baik dan koordinasi yang lebih efektif. Scrum memungkinkan tim untuk bekerja dalam sprint pendek yang terfokus, yang membantu menjaga momentum dan memastikan bahwa semua anggota tim tetap terlibat dan produktif. Dalam setiap sprint, tim menetapkan tujuan yang jelas dan dapat dicapai, yang membantu menjaga fokus dan mengurangi kebingungan tentang prioritas. Misalnya, penelitian oleh [8] menemukan bahwa sifat iteratif dan inkremental dari Scrum memungkinkan perusahaan untuk terus bersaing dalam meningkatkan produk mereka di pasar. Dengan adanya retrospektif di akhir setiap sprint, tim dapat mengevaluasi apa yang berhasil dan apa yang perlu diperbaiki, sehingga terus menerus meningkatkan proses kerja mereka.

Produktivitas juga ditingkatkan melalui pendekatan iteratif dan inkremental yang memecah proyek besar menjadi bagian-bagian yang lebih kecil dan lebih mudah dikelola. Hal ini tidak hanya membuat proyek lebih mudah untuk dikelola tetapi juga memungkinkan tim untuk memberikan nilai tambah lebih cepat dan lebih sering. Selain itu, fokus pada komunikasi yang kontinu memastikan bahwa setiap anggota tim selalu mengetahui tugas mereka dan bagaimana kontribusi mereka mendukung tujuan keseluruhan. Penggunaan agile dalam hal ini memungkinkan fleksibilitas dan responsif terhadap perubahan yang terjadi di pasar, meningkatkan daya saing perusahaan secara keseluruhan. Pendekatan agile dalam peningkatan produktivitas ini adalah penggunaan sprint yang terfokus dan iteratif, yang merupakan inti dari metodologi Agile untuk memastikan peningkatan produktivitas dan efisiensi tim.

4.4. Peningkatan Kualitas Produk

Scrum mendorong inspeksi dan adaptasi yang terus-menerus, yang pada akhirnya meningkatkan kualitas produk akhir. Dalam setiap sprint, tim melakukan review untuk mengevaluasi pekerjaan yang telah selesai dan memastikan bahwa produk memenuhi standar kualitas yang diinginkan. Retrospektif juga memberikan kesempatan bagi tim untuk mengidentifikasi area yang memerlukan perbaikan dan mengembangkan rencana untuk meningkatkan proses kerja di sprint berikutnya. Dengan melibatkan seluruh tim dalam proses review dan retrospektif, tim dapat belajar dari kesalahan dan terus memperbaiki proses kerja mereka.

Menurut [2], Scrum membantu tim mencapai kualitas perangkat lunak yang lebih tinggi melalui kombinasi kolaborasi tim, keselamatan psikologis, dan transparansi. Selain itu, dengan adanya peran Product

Owner yang bertanggung jawab untuk menjaga backlog produk dan memastikan bahwa setiap fitur yang dikembangkan sesuai dengan kebutuhan bisnis dan pengguna, kualitas produk dapat lebih terjamin.

Kualitas produk juga meningkat melalui umpan balik yang cepat dan berkelanjutan dari pemangku kepentingan. Dalam setiap sprint review, tim mendapatkan umpan balik langsung dari pemangku kepentingan yang memungkinkan mereka untuk segera menyesuaikan dan memperbaiki produk. Proses ini memastikan bahwa produk akhir lebih sesuai dengan kebutuhan dan harapan pengguna, serta memiliki lebih sedikit cacat. Penggunaan agile memungkinkan pengujian dan validasi terus menerus yang memperkuat kualitas produk secara keseluruhan. Pendekatan agile dalam peningkatan kualitas produk ini adalah proses inspeksi dan adaptasi terus-menerus serta umpan balik cepat dari pemangku kepentingan, yang merupakan praktik Agile untuk memastikan produk yang dihasilkan memiliki kualitas yang tinggi.

4.5. Peningkatan Kolaborasi Tim

Scrum mendorong inspeksi dan adaptasi yang terus-menerus, yang pada akhirnya meningkatkan kualitas produk akhir. Dalam setiap sprint, tim melakukan review untuk mengevaluasi pekerjaan yang telah selesai dan memastikan bahwa produk memenuhi standar kualitas yang diinginkan. Retrospektif juga memberikan kesempatan bagi tim untuk mengidentifikasi area yang memerlukan perbaikan dan mengembangkan rencana untuk meningkatkan proses kerja di sprint berikutnya. Dengan melibatkan seluruh tim dalam proses review dan retrospektif, tim dapat belajar dari kesalahan dan terus memperbaiki proses kerja mereka.

Menurut [2], Scrum membantu tim mencapai kualitas perangkat lunak yang lebih tinggi melalui kombinasi kolaborasi tim, keselamatan psikologis, dan transparansi. Selain itu, dengan adanya peran Product Owner yang bertanggung jawab untuk menjaga backlog produk dan memastikan bahwa setiap fitur yang dikembangkan sesuai dengan kebutuhan bisnis dan pengguna, kualitas produk dapat lebih terjamin.

Kualitas produk juga meningkat melalui umpan balik yang cepat dan berkelanjutan dari pemangku kepentingan. Dalam setiap sprint review, tim mendapatkan umpan balik langsung dari pemangku kepentingan yang memungkinkan mereka untuk segera menyesuaikan dan memperbaiki produk. Proses ini memastikan bahwa produk akhir lebih sesuai dengan kebutuhan dan harapan pengguna, serta memiliki lebih sedikit cacat. Penggunaan agile memungkinkan pengujian dan validasi terus menerus yang memperkuat kualitas produk secara keseluruhan. Pendekatan agile dalam peningkatan kualitas produk ini adalah proses inspeksi dan adaptasi terus-menerus serta umpan balik cepat dari pemangku kepentingan, yang merupakan praktik Agile untuk memastikan produk yang dihasilkan memiliki kualitas yang tinggi.

4.6. Adaptabilitas Terhadap Perubahan

Scrum memungkinkan tim untuk lebih adaptif terhadap perubahan kebutuhan dan kondisi proyek. Dalam lingkungan yang dinamis dan cepat berubah, kemampuan untuk beradaptasi dengan cepat sangat penting. Dengan sprint yang pendek dan fleksibilitas dalam mengatur backlog produk, tim dapat dengan mudah menyesuaikan prioritas dan fokus kerja mereka berdasarkan umpan balik terbaru dari pemangku kepentingan. Studi oleh [3] menunjukkan bahwa model hybrid antara Scrum dan Waterfall dapat mengoptimalkan komunikasi dan fleksibilitas, memungkinkan tim untuk menyesuaikan metode mereka dengan kebutuhan proyek yang spesifik.

Adaptabilitas ini juga ditingkatkan melalui pendekatan iteratif yang memungkinkan tim untuk menguji dan mengulang lebih cepat, memastikan bahwa produk yang dihasilkan selalu relevan dan memenuhi kebutuhan pengguna. Proses ini mempromosikan inovasi dan memungkinkan tim untuk merespons perubahan pasar dengan lebih efektif. Penggunaan agile dalam konteks ini memastikan bahwa tim selalu siap untuk menghadapi perubahan dan dapat beradaptasi dengan cepat untuk menjaga kelancaran dan keberhasilan proyek. Pendekatan agile dalam adaptabilitas terhadap perubahan ini adalah pendekatan iteratif dan fleksibilitas backlog, yang merupakan prinsip Agile untuk memastikan tim dapat beradaptasi dengan cepat terhadap perubahan kebutuhan dan kondisi proyek.

5. KESIMPULAN DAN SARAN

Pendekatan Agile dengan Scrum dalam tim pengembangan perangkat lunak terbukti meningkatkan transparansi, mengurangi hambatan komunikasi, meningkatkan produktivitas dan kualitas produk, serta memungkinkan adaptabilitas terhadap perubahan kebutuhan proyek. Untuk penelitian selanjutnya, disarankan untuk mengeksplorasi dampak jangka panjang dari Scrum terhadap kinerja tim dan keberhasilan proyek, serta mengembangkan strategi pembinaan budaya kolaborasi yang lebih efektif dan implementasi alat visualisasi untuk memperkuat transparansi dan koordinasi dalam tim.

DAFTAR PUSTAKA

- [1] F. Ramin, C. Matthies, dan R. Teusner, "More than Code: Contributions in Scrum Software Engineering Teams," dalam *Proceedings - 2020 IEEE/ACM 42nd International Conference on Software Engineering Workshops, ICSEW 2020*, Association for Computing Machinery, Inc, Jun 2020, hlm. 137–140. doi: 10.1145/3387940.3392241.
- [2] A. Alami dan O. Krancher, "How Scrum adds value to achieving software quality?," *Empir Softw Eng*, vol. 27, no. 7, Des 2022, doi: 10.1007/s10664-022-10208-4.
- [3] N. Yahya dan S. S. Maidin, "Hybrid agile development phases: The practice in software

- projects as performed by software engineering team,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 29, no. 3, hlm. 1738–1749, Mar 2023, doi: 10.11591/ijeecs.v29.i3.pp1738-1749.
- [4] A. Hidayati, E. K. Budiardjo, dan B. Purwandari, “Scrum team competence based on knowledge, skills, attitude in global software development,” *Quality - Access to Success*, vol. 22, no. 184, hlm. 93–98, Okt 2021, doi: 10.47750/QAS/22.184.11.
- [5] S. Shafiq, Y. Hafeez, S. Ali, N. Iqbal, dan M. Jamal, “Towards Scrum Based Agile Framework for Global Software Development Teams,” *Mehran University Research Journal of Engineering and Technology*, vol. 38, no. 4, hlm. 979–998, Okt 2019, doi: 10.22581/muet1982.1904.11.
- [6] Y. I. Alzoubi dan A. Q. Gill, “An Empirical Investigation of Geographically Distributed Agile Development: The Agile Enterprise Architecture is a Communication Enabler,” *IEEE Access*, vol. 8, hlm. 80269–80289, 2020, doi: 10.1109/ACCESS.2020.2990389.
- [7] L. Syahadiyanti, A. Raharja, dan Pamudi, “LITERATURE REVIEW: KOMUNIKASI SEBAGAI FAKTOR PENTING DALAM PENGEMBANGAN PERANGKAT LUNAK,” *JURNAL TEKNOLOGI TECHNOSCIENTIA*, hlm. 19–26, Apr 2023, doi: 10.34151/technoscientia.v15i2.4261.
- [8] M. Rizky dan Y. Sugiarti, “Pengunaan Metode Scrum Dalam Pengembangan Perangkat Lunak: Literature Review,” *Journal of Computer Science and Engineering (JCSE)*, vol. 3, no. 1, hlm. 41–48, Feb 2022, doi: 10.36596/jcse.v3i1.353.
- [9] T. Dingsøy, N. B. Moe, dan E. A. Seim, “Coordinating Knowledge Work in Multiteam Programs: Findings From a Large-Scale Agile Development Program,” *Project Management Journal*, vol. 49, no. 6, hlm. 64–77, Des 2018, doi: 10.1177/8756972818798980.
- [10] O. C. Resmi Rachmawati, Deyana Kusuma Wardani, Wifda Muna Fatihia, Arna Fariza, dan Hestiasari Rante, “Implementing Agile Scrum Methodology in The Development of SICITRA Mobile Application,” *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 7, no. 1, hlm. 41–50, Feb 2023, doi: 10.29207/resti.v7i1.4688.
- [11] S. Chaouch, A. Mejri, dan S. A. Ghannouchi, “A framework for risk management in Scrum development process,” dalam *Procedia Computer Science*, Elsevier B.V., 2019, hlm. 187–192. doi: 10.1016/j.procs.2019.12.171.
- [12] T. R. Ojha, “Critical Success Factors of Agile Software Development - A Systematic Literature Review,” *SCITECH Nepal*, vol. 17, no. 1, hlm. 49–57, Des 2023, doi: 10.3126/scitech.v17i1.60467.
- [13] P. Kruchten, S. Fraser, dan F. Coallier, *Agile Processes in Software Engineering and Extreme Programming*, vol. 355. dalam *Lecture Notes in Business Information Processing*, vol. 355. Cham: Springer International Publishing, 2019. doi: 10.1007/978-3-030-19034-7.
- [14] Y. I. Alzoubi dan A. Mishra, “Enterprise architecture contribution in distributed agile software development,” *Systems Engineering*, vol. 27, no. 3, hlm. 570–584, Mei 2024, doi: 10.1002/sys.21739.
- [15] B. Misnevs dan U. Demiray, “The Role of Communication and Meta-communication in Software Engineering with Relation to Human Errors,” dalam *Procedia Engineering*, Elsevier Ltd, 2017, hlm. 213–222. doi: 10.1016/j.proeng.2017.01.100.
- [16] M. D. Kadenic, D. A. de Jesus Pacheco, K. Koumaditis, G. Tjørnehøj, dan T. Tambo, “Investigating the role of Product Owner in Scrum teams: Differentiation between organisational and individual impacts and opportunities,” *Journal of Systems and Software*, vol. 206, Des 2023, doi: 10.1016/j.jss.2023.111841.
- [17] L. Barros, C. Tam, dan J. Varajão, “Agile software development projects—Unveiling the human-related critical success factors,” *Inf Softw Technol*, vol. 170, Jun 2024, doi: 10.1016/j.infsof.2024.107432.