

INTEGRASI METODE SCRUM DAN DEVOPS PADA PENGEMBANGAN SISTEM INFORMASI UKM MULTIMEDIA AISKA UNIVERSITY

Irfan Sadida¹, Ita Permatahati^{2*}, Nalurita Enggar Renosih³, Faiq Fadhil Dzulfihar Bariq⁴

^{1,2,3,4} Fakultas Sains dan Teknologi, Universitas 'Aisyiyah Surakarta, *Penulis Koresponden
e-mail: ¹irfansadida.students@aiska-university.ac.id, ²itapermata02@aiska-university.ac.id,
³naluritaer.students@aiska-university.ac.id, ⁴faiqfadhil.students@aiska-university.ac.id

ABSTRACT

The increasing need for information systems by organizations, especially in managing resources efficiently, drives the importance of implementing effective software development methods. This research discusses the integration of the Scrum method and the DevOps method in a web-based information system development project for multimedia SMEs at Aiska University (MUSKA). This integration aims to utilize the advantages of the Scrum and DevOps methods to optimize the information system development process. The integration of the two methods is assessed through team performance, by identifying the benefits and challenges faced in its implementation. This research shows that collaboration between Scrum and DevOps results in a more effective and reliable development process, because each method complements each other in overcoming MUSKA system development challenges. Projects can be managed more systematically, and automation allows for faster responses to changes and increased quality of the final output. The evaluation of this research shows that further optimization of the roles and functions of each team member requires an increase in project resources, both in terms of adding team members and using more advanced technology. These findings provide recommendations for future development, especially for large-scale information system development projects.

Keywords: Scrum, DevOps, MUSKA, Laravel, Website

INTISARI

Peningkatan kebutuhan sistem informasi oleh organisasi, khususnya dalam mengelola sumber daya secara efisien, mendorong pentingnya penerapan metode pengembangan perangkat lunak yang efektif. Penelitian ini membahas tentang integrasi dari metode Scrum dan metode DevOps dalam proyek pengembangan sistem informasi berbasis web untuk UKM multimedia di Aiska University (MUSKA). Integrasi ini bertujuan untuk memanfaatkan keunggulan dari metode scrum dan DevOps untuk mengoptimalkan proses pengembangan sistem informasi. Integrasi kedua metode dinilai melalui kinerja tim, dengan mengidentifikasi keuntungan dan tantangan yang dihadapi dalam penerapannya. Penelitian ini menunjukkan bahwa kolaborasi antara Scrum dan DevOps menghasilkan proses pengembangan yang lebih efektif dan dapat diandalkan, karena setiap metode saling melengkapi dalam mengatasi tantangan pengembangan sistem MUSKA. Proyek dapat dikelola dengan lebih sistematis, dan otomatisasi memungkinkan respons yang lebih cepat terhadap perubahan serta peningkatan kualitas hasil akhir. Adapun evaluasi dari penelitian ini yaitu menunjukkan bahwa optimalisasi lebih lanjut peran dan fungsi setiap anggota tim dan memerlukan peningkatan sumber daya proyek, baik dalam hal penambahan anggota tim maupun penggunaan teknologi yang lebih maju. Temuan ini memberikan rekomendasi untuk pengembangan di masa yang akan datang, khususnya untuk proyek pengembangan sistem informasi berskala besar.

Kata kunci: Scrum, DevOps, MUSKA, Laravel, Website

1. PENDAHULUAN

Di era digital saat ini, kebutuhan akan ketersediaan informasi yang akurat dan cepat semakin meningkat. Ketersediaan informasi yang tepat dapat mendukung aktifitas organisasi dalam mengelola sumber daya, mengambil keputusan strategis, serta meningkatkan efisiensi kinerja operasional. Dengan sistem informasi yang terpusat, organisasi dapat merencanakan tindakan dengan lebih baik, mengidentifikasi potensi dan tantangan, serta menyesuaikan perubahan kebutuhan bisnis yang dinamis.

Metode pengembangan sistem adalah kunci dalam menghasilkan proyek yang berpengaruh pada penentuan keberhasilan pengembangan terhadap peran setiap anggota tim (Olorunshola & Ogwueleka, 2022). Metode pengembangan sistem yang efisien dan dapat diandalkan menjadi hal yang perlu ditentukan dalam proses pengembangan perangkat lunak. Perangkat lunak yang dihasilkan harus dapat memenuhi kebutuhan pengguna yang dinamis. Dalam lingkungan yang kompetitif saat ini, metode pengembangan perangkat lunak tradisional sering kali mendapatkan keterbatasan, seperti lambatnya waktu respon terhadap feedback pengguna yang disebut permasalahan otomatisasi dan kurangnya kolaborasi antar tim yang disebut permasalahan manajemen tim.

Metode Scrum merupakan metode yang banyak diadopsi oleh perusahaan untuk meningkatkan efisiensi dan kolaborasi dalam tim pengembangan perangkat lunak. Scrum menawarkan kerangka kerja yang berbasis iterasi dan kolaborasi untuk meningkatkan produktivitas tim pengembang. Menurut (Alsaqqa dkk., 2020) metode scrum memiliki keunggulan dalam mengorganisasi tim pengembang melalui pembagian tugas yang jelas, komunikasi intensif antar anggota tim, dan akomodasi kebutuhan bisnis melalui feedback pengguna. Dengan menekankan pada pembagian role yang jelas pada setiap *task* dan siklus pengembangan yang pendek, Scrum memungkinkan tim pengembang untuk beradaptasi dengan cepat terhadap perubahan kebutuhan pengguna yang dinamis. Sedangkan, Metode DevOps muncul untuk mempercepat siklus pengembangan perangkat lunak dan meningkatkan kolaborasi antara pengembang dan tim operasional. DevOps merupakan suatu pendekatan yang digunakan dalam pengembangan perangkat lunak dengan menselaraskan antara lingkungan *development* dan lingkungan *operation*. (Almeida dkk., 2022) mengidentifikasi metode DevOps memiliki keunggulan dalam otomatisasi, kualitas code, penanggulangan resiko pada lingkungan *operation* dan visibilitas perubahan code. Dengan menerapkan prinsip-prinsip DevOps, proses pengembangan sistem dapat dilakukan secara lebih harmonis, mengurangi batasan antara tim melalui otomatisasi, dan mempercepat siklus pengembangan.

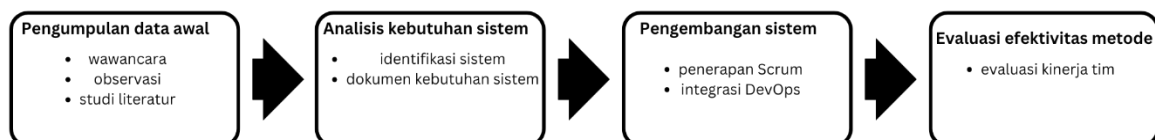
Berbagai penelitian terkait penerapan Scrum dan DevOps dalam pengembangan perangkat lunak telah dilakukan. Misalnya, penelitian yang dilakukan oleh (Muay dkk., 2024) yang menunjukkan bahwa penerapan metode Scrum memberikan proses manajemen produk yang terstruktur dan tugas yang terselesaikan. Penelitian lain oleh (Hemon dkk., 2020) mengungkapkan bahwa penerapan DevOps pada tim dapat meningkatkan kolaborasi antar tim pengembang dan operasi, sehingga meminimalisir kegagalan dan kesalahan dalam pengiriman perangkat lunak.

Penelitian ini bertujuan untuk mengintegrasikan metode scrum dan DevOps. Integrasi kedua metode dilakukan untuk mendapatkan metode pengembangan sistem yang efektif melalui keunggulan dari kedua metode tersebut. Integrasi metode Scrum dan DevOps diimplementasikan dalam ruang lingkup terbatas pada pengembangan sistem informasi UKM Multimedia Universitas Aisyiyah Surakarta sehingga efektifitas dapat diukur secara spesifik. Sistem ini yang berikutnya disebut dengan MUSKA dirancang sebagai langkah strategis untuk menyediakan informasi yang terpusat mengenai UKM Multimedia Universitas Aisyiyah Surakarta. Sistem MUSKA dapat memberikan kemudahan bagi pengurus UKM Multimedia dalam mengelola data dan informasi kepada mahasiswa secara efektif. Sehingga dapat mendukung aktifitas mahasiswa dalam mengembangkan keterampilan yang dimiliki.

2. METODE PENELITIAN

2.1. Desain Penelitian

Dalam penelitian ini, pendekatan kualitatif deskriptif dengan studi kasus pada proyek pengembangan sistem informasi manajemen UKM Multimedia (MUSKA) berbasis web menggunakan framework Laravel. Secara terperinci, alur penelitian disajikan dalam diagram berikut ini.



Gambar 1. Diagram Alur Penelitian

Pengumpulan data dilakukan melalui wawancara, observasi, dan studi dokumentasi. Wawancara dilakukan dengan Product Owner, yang dalam hal ini diwakili oleh pembimbing UKM Multimedia, untuk menggali kebutuhan dan tujuan pengembangan sistem. Selanjutnya, observasi dilakukan pada berbagai sistem informasi organisasi yang tersedia di internet, untuk memperoleh referensi terkait struktur sistem yang akan dikembangkan. Selain itu, studi literatur dilakukan dengan mengkaji artikel, buku, dan karya ilmiah yang relevan, sebagai dasar teori dalam pengembangan sistem. Hasil dari pengumpulan data tersebut akan dianalisis untuk menyusun Product Backlog, yang akan menjadi acuan dalam pengembangan sistem. Proses pengembangan sistem selanjutnya dilakukan dengan menerapkan Scrum sebagai kerangka kerja manajemen proyek berbasis iterasi, serta

mengintegrasikan DevOps untuk otomatisasi dalam seluruh siklus pengembangan. Pada tahap akhir, dilakukan evaluasi terhadap kinerja tim dalam pengembangan sistem, guna mengetahui efektivitas integrasi metode Scrum dan DevOps dalam meningkatkan kualitas dan efisiensi pengembangan sistem.

2.2. Alat

Penelitian ini memanfaatkan berbagai alat pendukung untuk meningkatkan kemampuan tim dalam menyelesaikan tugas-tugas tertentu. Figma berfungsi sebagai software perancang desain antarmuka sistem dan diagram analisis sistem. Google Spreadsheet dan Docs berfungsi untuk pencatatan dokumen scrum. Google Meet dan Zoom digunakan sebagai software pertemuan online yang memfasilitasi tim scrum, master scrum, dan stakeholder dalam berdiskusi secara daring mengenai kemajuan pengembangan sistem. Untuk mengelola berbagai tugas dalam pengembangan sistem selama sprint berlangsung, Clickup digunakan sebagai platform dengan fitur-fitur yang memudahkan dalam manajemen tugas dan memonitoring progres pengembangan sistem (Widiyanti & Nugroho, 2024).

Selain alat pendukung manajemen proyek dan komunikasi, Laravel dimanfaatkan sebagai framework utama dalam pengembangan sistem MUSKA. Laravel dipilih karena kemampuannya dalam membangun aplikasi web yang scalable, aman dan mudah dikembangkan. Sebagai framework PHP Laravel juga mendukung penerapan alat pengembangan modern, seperti pengujian otomatis dan *continuous integration / continuous delivery (CI/CD)* (Hany Fawzy dkk., 2023).

2.3. Metode Pengembangan

Integrasi metode Scrum dan DevOps diterapkan bertujuan untuk memanfaatkan kekuatan masing-masing metode guna mendapatkan metode pengembangan yang efektif dan dapat diandalkan. Adapun Scrum digunakan untuk menyediakan kerangka kerja yang iteratif dan kolaboratif, Sedangkan DevOps digunakan untuk otomatisasi dalam proses pengembangan serta mempercepat siklus pengiriman perangkat lunak. Penggabungan kedua metode tersebut dilakukan berdasarkan peninjauan literatur dan penelitian yang mengarah pada kemungkinan dalam mengintegrasikan metode Scrum dan DevOps (Almeida dkk., 2022; Alsaqqa dkk., 2020).

Penerapan metode Scrum dalam pengembangan sistem MUSKA mengikuti alur yang terstruktur, dengan lima agenda utama: Sprint, Sprint Planning, Daily Scrum, Sprint Review, dan Sprint Retrospective. Sebagai langkah awal, analisis sistem dilakukan melalui wawancara dengan Product Owner dan observasi dokumentasi proyek, yang menghasilkan dokumen perencanaan proyek atau Product Backlog. Sprint diawali dengan Sprint Planning untuk menetapkan tugas yang akan dikerjakan, berdasarkan Product Backlog yang telah disempurnakan. Tim Scrum kemudian membuat Sprint Backlog yang berisi tugas-tugas yang akan diselesaikan selama Sprint (Verheyen, 2024).

Selama Sprint, Daily Scrum dilaksanakan untuk memantau kemajuan tim dan memastikan bahwa setiap tugas sesuai rencana. Dalam tahap ini, prinsip DevOps diintegrasikan untuk mempercepat siklus pengembangan. DevOps mengutamakan kolaborasi antara tim pengembangan dan operasi, serta mengoptimalkan penggunaan alat otomatisasi seperti Continuous Integration (CI) dan Continuous Delivery (CD). Dengan DevOps, perubahan kode dapat segera diuji dan diterapkan tanpa menunggu tahap rilis panjang, yang mempercepat waktu feedback dan mengurangi kesalahan antara pengembangan dan produksi (Azad & Hyrynsalmi, 2023).

Hasil kemajuan sprint dipresentasikan dalam Sprint Review untuk dievaluasi oleh tim dan stakeholder. Sprint ditutup dengan Sprint Retrospective, di mana tim Scrum melakukan evaluasi internal untuk meningkatkan kualitas kerja dalam iterasi berikutnya (Eryc, 2021).

Integrasi metode Scrum dan DevOps bertujuan mengoptimalkan efisiensi, (Azad & Hyrynsalmi, 2023) meningkatkan kualitas produk, dan memastikan pengembangan berjalan lebih terstruktur. Penerapan metodologi ini diharapkan menghasilkan sistem yang lebih stabil dan dapat diandalkan, dengan proses yang lebih efisien dan terdokumentasi dengan baik. Alur kerangka kerja integrasi metode Scrum dan DevOps dapat dilihat pada **Gambar 2**.



Gambar 2. Integrasi Metode Scrum dan DevOps

3. HASIL DAN PEMBAHASAN

3.1. Hasil

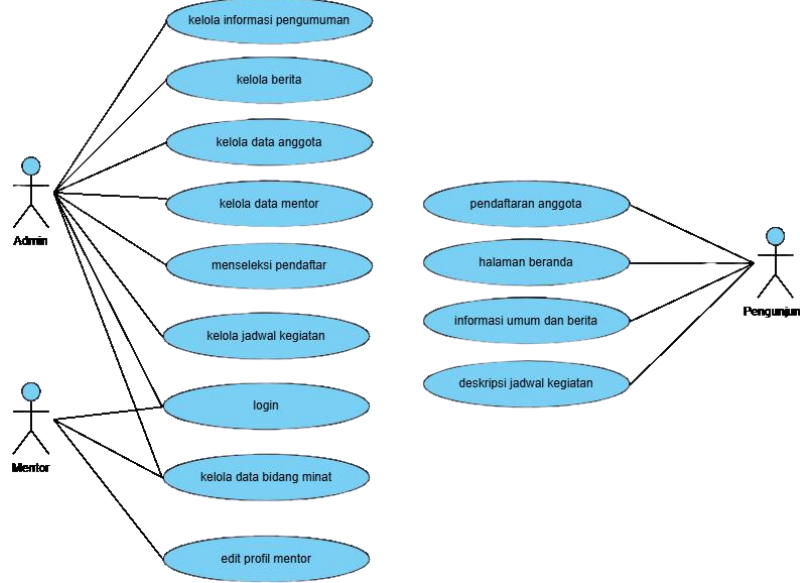
Penelitian ini menerapkan pendekatan integrasi antara metode Scrum dan DevOps dengan mengikuti prinsip-prinsip yang ada dalam kerangka kerja yang telah ditentukan. Berbagai dokumentasi dan visual juga disertakan untuk menjelaskan integrasi penerapan kedua metodologi dalam pengembangan sistem MUSKA. Pada tahap awal tim pengembang berkolaborasi dengan product owner dan stakeholder Menyusun dokumen perencanaan proyek melalui wawancara langsung. Dokumen tersebut berisi tentang tujuan pembangunan sistem, fungsi sistem, spesifikasi sistem, sumber daya, elisitasi kebutuhan dan kontrak kerja. **Tabel 1** menyajikan hasil elisitasi kebutuhan sistem yang diperoleh dari wawancara dengan product owner dan stakeholder. Dokumen tersebut diuraikan menjadi product backlog yang akan menjadi acuan tim pengembang dalam proses pengembangan sistem.

Tabel 1. Elisitasi Kebutuhan Sistem MUSKA

| No | Kebutuhan | Keterangan |
|-----|--|------------------|
| 1. | From pendaftaran online bagi calon anggota baru ukm multimedia | Functional |
| 2. | Deskripsi masing-masing bidang minat ukm multimedia | Functional |
| 3. | Jadwal kegiatan pelatihan | Functional |
| 4. | Jadwal kegiatan event | Functional |
| 5. | Berita update ukm multimedia | Functional |
| 6. | Harus ada halaman admin | Functional |
| 7. | Dapat mengelola data anggota | Functional |
| 8. | Dapat mengelola data event | Functional |
| 9. | Dapat mengelola data pengumuman | Functional |
| 10. | Dapat mengelola data bidang minat | Functional |
| 11. | Dapat mengelola data mentor | Functional |
| 12. | Ada halaman akses bagi pengurus dan admin | Functional |
| 13. | Ada menu log-out | Functional |
| 14. | Ada menu log-in | Functional |
| 15. | Ada halaman log-in bagi admin | Functional |
| 16. | Ada halaman log-out bagi admin | Functional |
| 17. | Ada halaman log-in bagi mentor | Functional |
| 18. | Website bersifat online | Functional |
| 19. | Ada pelaporan | Functional |
| 20. | Menampilkan data pengurus ukm multimedia | Functional |
| 21. | Menampilkan alamat, no.telp, email | Functional |
| 22. | Website mudah dipahami | Non - Functional |
| 23. | Warna tampilan website sesuai dengan warna logo | Non - Functional |

Pada **Gambar 3** dibawah ini mengilustrasikan *use case diagram* interaksi setiap aktor dalam sistem MUSKA. Use case diagram merupakan Bahasa modeling yang digunakan untuk menggambarkan interaksi antara sistem dan aktor eksternal (Arifin & Siahaan, 2020). Pengunjung dapat melakukan berbagai aktivitas, seperti mendaftar sebagai anggota, melihat jadwal kegiatan, membaca berita, dan mengakses informasi umum lainnya. Sementara itu role admin dan mentor memerlukan proses login. Admin memiliki wewenang untuk mengelola mentor, jadwal, berita, data anggota, seleksi pendaftaran, serta mengelola data bidang minat dan pelaporan. Sedangkan, mentor dapat melakukan edit profil serta mengelola data terkait bidang minat. Proses pengembangan sistem akan dibagi menjadi tiga sprint, setiap sprint dijadwalkan berangsur selama satu minggu. Selanjutnya setiap sprint tim pengembang dapat berfokus pada penyelesaian tugas-tugas yang telah ditentukan. Pada gambar 4 menampilkan sprint backlog yang disusun sebagai item product backlog dan diselesaikan secara bertahap. Pembagian peran dalam pengembangan sistem MUSKA diuraikan dalam gambar 5. Bapak Sigit sebagai *product owner* yang mewakili pembimbing UKM Multimedia Aiska. Sebagai *product owner* berkolaborasi dengan tim pengembang dalam menyusun dokumen perencanaan proyek dan mengevaluasi kemajuan pengembangan. Dalam lingkup tim scrum ibu Ita sebagai scrum master mengawasi efisiensi tim serta menavigasi tim dalam mengatasi hambatan dan

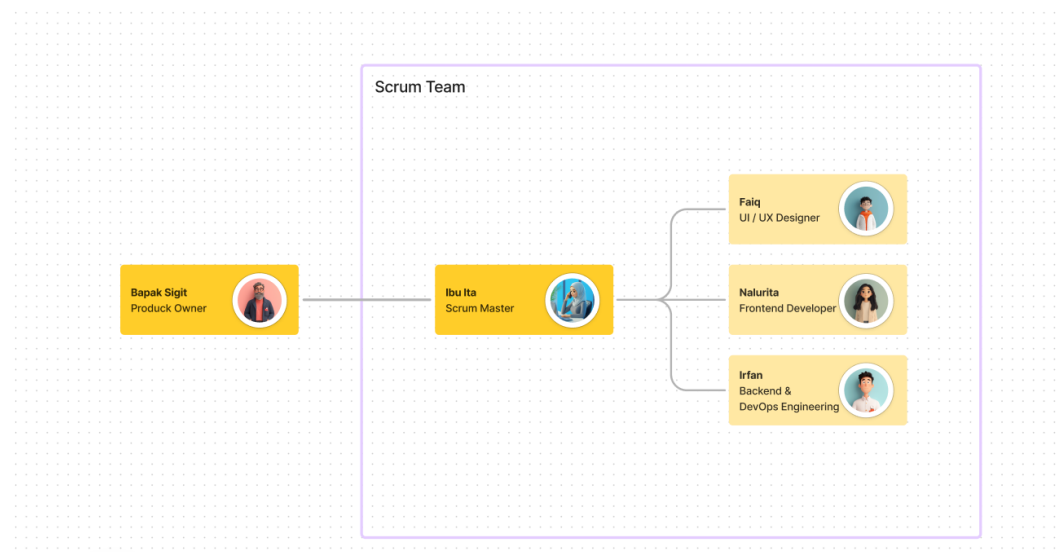
konflik tanpa keterlibatan langsung secara teknis. Lalu disisi teknis Faiq Fadhill sebagai UI/UX desainer, Nalurita sebagai Frontend Developer, dan Irfan Sadida sebagai Backend & DevOps Engineering.



Gambar 3. Use Case Diagram



Gambar 4. Product Backlog Web MUSKA

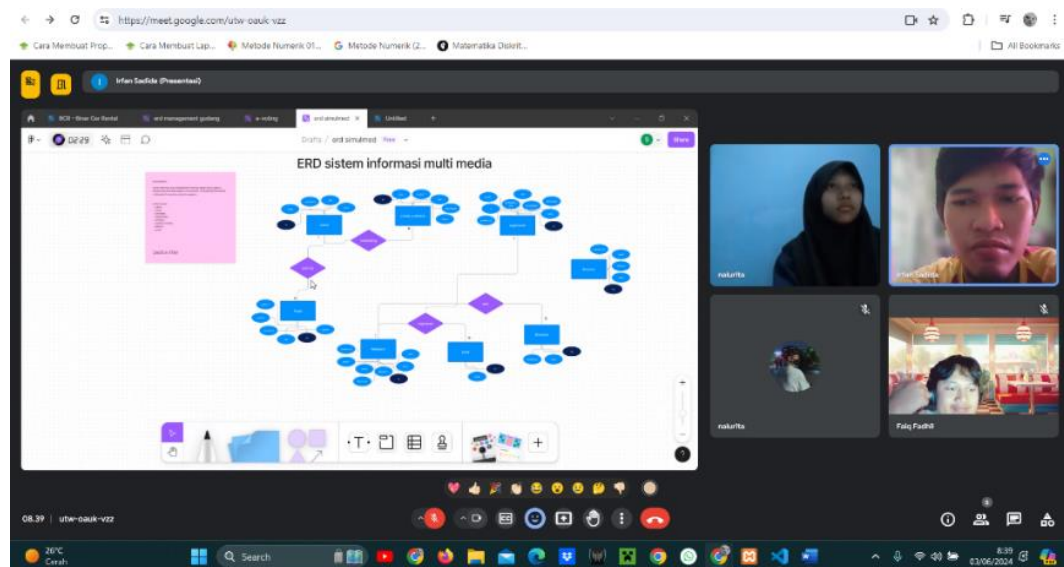


Gambar 5. Pembagian Peran Aktor Scrum

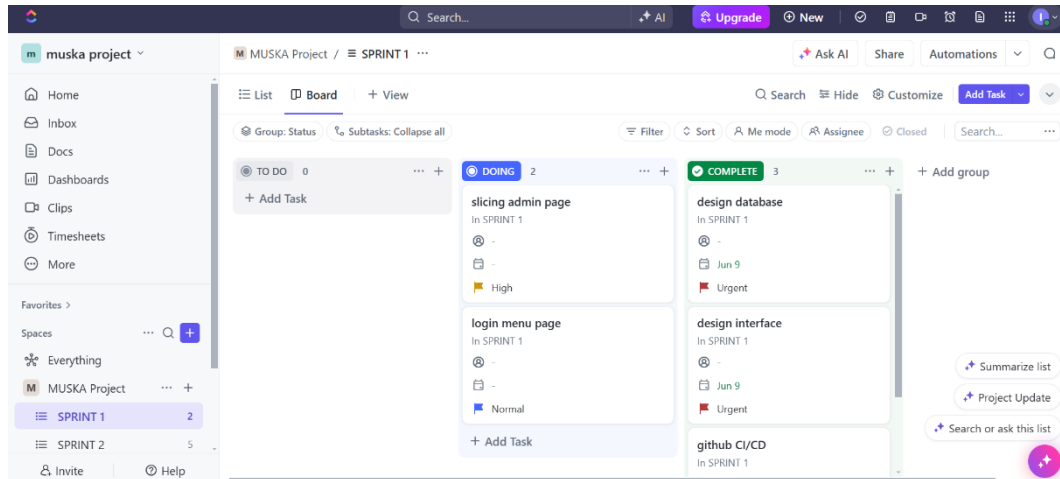
Selama *daily scrum*, tim *scrum* mengadakan pertemuan setiap pagi pada pukul 08.30 selama 15 menit secara konsisten. Pertemuan ini dihadiri oleh seluruh anggota tim *scrum* untuk melakukan koordinasi dan melaporkan progres penyelesaian tugas yang telah ditentukan. Gambar 6 menunjukkan salah satu kegiatan *daily meeting* yang dilakukan oleh tim *scrum* secara rutin menggunakan Google Meet. Tim *scrum* memanfaatkan ClickUp sebagai alat manajemen tugas, di mana setiap anggota tim dapat memantau dan memperbarui status tugas yang sudah selesai atau masih dalam pengerjaan. Manajemen status tugas ini dapat dilihat pada Gambar 7. Selain itu github action diterapkan oleh tim teknis untuk melakukan pengujian otomatisasi dan CI/CD sehingga proses rilis produk dapat di evaluasi lebih cepat oleh product owner dan stakeholder.

Pada 9 Juli 2024, diadakan *daily scrum* di mana setiap anggota tim melaporkan hal-hal yang meliputi apa yang telah dikerjakan, kendala atau hambatan yang dihadapi, serta apa yang akan dikerjakan hari itu. Irfan melaporkan telah merancang struktur database dan berencana mengimplementasikan rancangan database pada hari itu. Faiq melaporkan telah merancang interface halaman admin dan berencana memperbaiki desain interface untuk pengunjung. Sedangkan Nalurita melaporkan telah melakukan slicing pada sebagian halaman interface admin dan akan menyelesaikannya. Selama *daily scrum* berlangsung, tidak ada anggota tim yang melaporkan kendala atau hambatan yang dihadapi pada hari itu.

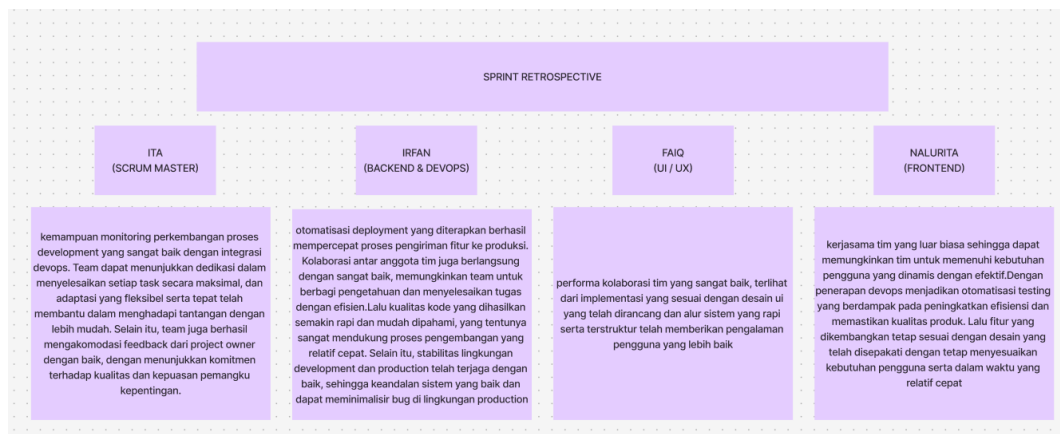
Pada *sprint rievew*, tim *scrum* memaparkan capaian mereka kepada product owner yang berpartisipasi dalam diskusi dan penilaian kemajuan menuju *sprint backlog*. *Sprint* diakhiri dengan acara *sprint retrospective*, di mana tim *scrum* melakukan evaluasi terhadap kinerja yang telah berlangsung. *Retrospective* membahas apa yang perlu dipertahankan, tantangan yang dihadapi dan hal yang perlu diperbaiki untuk *sprint* berikutnya. Durasi *sprint retrospective* disesuaikan dengan kebutuhan tim dalam mengevaluasi kinerja tim. Gambar 8 menampilkan tanggapan *retrospective* dari setiap anggota tim *scrum* diakhir *sprint*.



Gambar 6. Daily meeting oleh Tim Scrum



Gambar 7. Manajemen Status Tugas dengan Clickup



Gambar 8. Sprint Retrospective

3.2. Pembahasan

Berdasarkan hasil penelitian ini, terdapat berbagai keuntungan dari integrasi metode Scrum dan DevOps dalam pengembangan perangkat lunak. Keuntungan integrasi kedua metode diperinci pada **tabel 2**.

Tabel 2. Keuntungan Integrasi Metode Scrum dan DevOps pada Pengembangan Sistem MUSKA

| Keuntungan | Hasil |
|-------------------------|---|
| Kualitas code | Standarisasi kode pada DevOps memberikan peringatan terkait gaya penulisan yang tidak sesuai, memastikan bahwa kode ditulis sesuai dengan aturan yang ditetapkan sehingga menghasilkan kode yang rapi dan konsisten |
| Kecepatan pengembangan | Meningkatkan kecepatan pengembangan dengan penerapan siklus iteratif yang terstruktur pada Scrum, serta otomatisasi dan <i>Continuous Integration</i> (CI) pada DevOps. Tim scrum dapat mengidentifikasi dan mengatasi hambatan yang dihadapi dengan tetap menyesuaikan kebutuhan bisnis. |
| Kesesuaian desain | Kolaborasi yang intensif antar anggota tim melalui daily scrum menghasilkan desain yang adaptif dan relevan dengan pengguna. |
| Performa kolaborasi tim | Metode Scrum meningkatkan kolaborasi tim dengan daily scrum yang memastikan komunikasi antara anggota tim. Setiap capaian tugas beserta hambatan disampaikan oleh setiap anggota dalam agenda daily meeting |
| Feedback pengguna | Dengan penerapan CI/CD metode DevOps product owner dan stakeholder dapat mengamati kemajuan tim pengembang secara langsung, sehingga umpan balik dari stakeholder dapat diterapkan lebih cepat dan efisien |

| Keuntungan | Hasil |
|---------------------------------|---|
| Perubahan kebutuhan produk | Scrum memungkinkan tim untuk beradaptasi dengan perubahan kebutuhan product melalui sprint yang bersifat fleksibel dan dapat disesuaikan. |
| Otomatisasi | DevOps memfasilitasi otomatisasi dalam pengujian, integrasi, dan pengiriman perangkat lunak yang akan mengurangi kesalahan manual dan meningkatkan efisiensi. Otomatisasi dapat menguatkan kemampuan tim dalam menerima kemungkinan perubahan kebutuhan. |
| Stabilitas dan keandalan sistem | Dengan pengujian otomatis secara terus menerus serta penerapan prinsip CI/CD, Tim pengembang dapat mendeteksi dan memperbaiki bug secara lebih cepat, meminimalisir kesalahan di lingkungan produksi, dan memastikan sistem berjalan secara stabil serta dapat diandalkan |

Berdasarkan **Tabel 2** tentang integrasi metode Scrum dan DevOps pada pengembangan sistem MUSKA menghasilkan 8 keuntungan yang signifikan. Integrasi kedua metode tersebut menunjukkan suatu metode pengembangan perangkat lunak yang efisien dan dapat diandalkan. Sifat saling melengkapi dari kedua metode pengembangan dibuktikan dalam analisis yang menyeluruh pada kerangka kerja metode Scrum dan DevOps yang saling menguatkan. Scrum memfasilitasi kerangka kerja iterative untuk manajemen tugas tim, sementara DevOps diterapkan dalam proses pengembangan oleh tim scrum dengan mengandalkan otomatisasi tugas. Keberhasilan integrasi metode Scrum dan DevOps.

4. KESIMPULAN

Berdasarkan hasil pembahasan pengembangan sistem informasi UKM multimedia Universitas 'Aisyiyah Surakarta menggunakan integrasi metode Scrum dan DevOps dapat ditarik Kesimpulan bahwa (1) integrasi metode Scrum dan DevOps dapat memaksimalkan kedua metode dengan keunggulan pada kerangka kerja yang disediakan oleh setiap metode. (2) Proses manajemen proyek dapat terstruktur dan didukung oleh setiap kegiatan yang terdokumentasi dengan baik. (3) Otomatisasi pada pengujian, integrasi dan pengiriman perangkat lunak secara signifikan meningkatkan kecepatan proses pengembangan dan menghindari kesalahan manual. (4) Integrasi kedua metode menghasilkan metode pengembangan sistem yang efektif dan dapat diandalkan. Evaluasi yang kami peroleh dari penelitian ini untuk masa mendatang adalah kebutuhan penambahan sumber daya proyek baik anggota tim ataupun penggunaan teknologi untuk meningkatkan peran dan fungsi setiap anggota tim.

UCAPAN TERIMA KASIH

Penelitian ini didukung oleh Universitas 'Aisyiyah Surakarta melalui bantuan dana penelitian dengan nomor kontrak Nomor 048/PN/III/2024 tahun anggaran 2024, sehingga menguatkan penulis untuk melakukan pengamatan yang lebih mendalam untuk menguji efektivitas integrasi metode Scrum dan DevOps, yang diharapkan dapat meningkatkan kualitas proyek pengembangan sistem informasi berbasis web.

DAFTAR PUSTAKA

- Almeida, F., Simões, J., & Lopes, S. (2022). Exploring the Benefits of Combining DevOps and Agile. *Future Internet*, 14(2), Article 2. <https://doi.org/10.3390/fi14020063>
- Alsaqqa, S., Sawalha, S., & Abdel-Nabi, H. (2020). Agile Software Development: Methodologies and Trends. *International Journal of Interactive Mobile Technologies (IJIM)*, 14(11), 246. <https://doi.org/10.3991/ijim.v14i11.13269>
- Arifin, M. N., & Siahaan, D. (2020). Structural and semantic similarity measurement of UML use case diagram. *Lontar Komputer: Jurnal Ilmiah Teknologi Informasi*, 11(2), 88.
- Azad, N., & Hyrynsalmi, S. (2023). DevOps critical success factors—A systematic literature review. *Information and Software Technology*, 157, 107150. <https://doi.org/10.1016/j.infsof.2023.107150>
- Eryc, E. (2021). Perancangan dan Implementasi Aplikasi Sistem Pendaftaran Sidang KP, Skripsi dan Tesis Online dengan Metode Scrum. *Journal of Information System and Technology (JOINT)*, 2(2), 49–55. <https://doi.org/10.37253/joint.v2i2.6086>
- Hany Fawzy, A., Wassif, K., & Moussa, H. (2023). Framework for automatic detection of anomalies in DevOps. *Journal of King Saud University - Computer and Information Sciences*, 35(3), 8–19. <https://doi.org/10.1016/j.jksuci.2023.02.010>
- Hemon, A., Lyonnet, B., Rowe, F., & Fitzgerald, B. (2020). From Agile to DevOps: Smart Skills and Collaborations. *Information Systems Frontiers*, 22(4), 927–945. <https://doi.org/10.1007/s10796-019-09905-1>
- Muay, N. T., Sedyono, E., & Tambotih, J. (2024). Integration Waterfall and Scrum Methodology in The Development of SIMARGA Web Application. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi*

- Informasi*, 8(2), Article 2. <https://doi.org/10.29207/resti.v8i2.5652>
- Olorunshola, O. E., & Ogwueleka, F. N. (2022). Review of System Development Life Cycle (SDLC) Models for Effective Application Delivery. Dalam A. Joshi, M. Mahmud, R. G. Ragel, & N. V. Thakur (Ed.), *Information and Communication Technology for Competitive Strategies (ICTCS 2020)* (hlm. 281–289). Springer. https://doi.org/10.1007/978-981-16-0739-4_28
- Verheyen, G. (2024). *Scrum: A Pocket Guide*. 1–100.
- Widiyanti, S., & Nugroho, S. P. (2024). *Scrum Implementation using ClickUp on Rentors Website Application Development*. 154–162. https://doi.org/10.2991/978-94-6463-366-5_15