



Optimasi Proses Bisnis Akuntansi Melalui Tahapan System Development Lifecycle Yang Efisien

O.Feriyanto¹, Riska Saumira Amanda², Citra Rahayu³, Lusiana Lusiana⁴, Okeu Kusmayanti⁵, Shalma Ayu Miranda P⁶

¹⁻⁶Universitas Teknologi Digital

Abstract. *This research focuses on optimizing accounting business processes through efficient stages of the System Development Lifecycle (SDLC). SDLC is a framework used to plan, develop, test, and maintain accounting information systems. In this study, each stage of the SDLC is analyzed to identify steps that can enhance the efficiency and effectiveness of accounting systems. The research method used is SDLC with a Waterfall approach, involving the stages of planning, analysis, design, implementation, testing, and maintenance. The results show that applying a structured SDLC can help companies manage accounting data more effectively, increase accuracy, reduce errors, and speed up accounting processes. This optimization also includes selecting the appropriate SDLC model, such as Waterfall and Prototype, based on the specific needs of the project and business environment. Thus, this research is expected to provide practical guidance for companies in optimizing their accounting business processes.*

Keywords: *Accounting Business Process, System Development Lifecycle (SDLC), Efficiency, Waterfall, Prototype.*

Abstrak. Penelitian ini berfokus pada optimasi proses bisnis akuntansi melalui tahapan System Development Lifecycle (SDLC) yang efisien. SDLC adalah kerangka kerja yang digunakan untuk merencanakan, mengembangkan, menguji, dan memelihara sistem informasi akuntansi. Dalam penelitian ini, setiap tahap SDLC dianalisis untuk mengidentifikasi langkah-langkah yang dapat meningkatkan efisiensi dan efektivitas sistem akuntansi. Metode penelitian yang digunakan adalah SDLC dengan pendekatan Waterfall, yang melibatkan tahapan perencanaan, analisis, desain, implementasi, pengujian, dan pemeliharaan. Hasil penelitian menunjukkan bahwa penerapan SDLC yang terstruktur dapat membantu perusahaan mengelola data akuntansi lebih efektif, meningkatkan akurasi, mengurangi kesalahan, dan mempercepat proses akuntansi. Optimasi ini juga mencakup pemilihan model SDLC yang tepat, seperti Waterfall dan Prototype, berdasarkan kebutuhan spesifik proyek dan lingkungan bisnis. Dengan demikian, penelitian ini diharapkan dapat memberikan panduan praktis bagi perusahaan dalam mengoptimalkan proses bisnis akuntansi mereka.

Kata kunci: Proses Bisnis Akuntansi, System Development Lifecycle (SDLC), Efisiensi, Waterfall, Prototype

PENDAHULUAN

Proses bisnis akuntansi merupakan tulang punggung operasional keuangan dalam sebuah organisasi. Keberhasilan suatu perusahaan seringkali bergantung pada efisiensi dan efektivitas proses akuntansi yang diterapkan (Romney & Steinbart, 2018, hal. 45). Namun, banyak perusahaan menghadapi tantangan dalam mengelola proses ini secara optimal. Proses yang tidak efisien dapat mengakibatkan kesalahan, penundaan, dan biaya operasional yang lebih tinggi (Romney & Steinbart, 2018, hal. 47).

Untuk mengatasi masalah ini, diperlukan pendekatan sistematis dalam pengembangan dan pemeliharaan sistem akuntansi. Salah satu pendekatan yang banyak digunakan adalah *System Development Lifecycle (SDLC)* (Kendall & Kendall, 2019, hal. 67). *SDLC* merupakan kerangka kerja yang membantu dalam perencanaan, pengembangan, pengujian, dan pemeliharaan sistem informasi (Kendall & Kendall, 2019, hal. 70). Melalui tahapan-tahapan

Received Juni 30, 2024; Accepted Juli 01, 2024; Published Juli 31, 2024

* O.Feriyanto

SDLC yang efisien, perusahaan dapat memastikan bahwa sistem akuntansi yang dikembangkan memenuhi kebutuhan bisnis, meningkatkan akurasi, mengurangi kesalahan, dan mempercepat proses akuntansi (Romney & Steinbart, 2018, hal. 48).

Dalam buku "*Accounting Information Systems*" oleh Marshall B. Romney dan Paul J. Steinbart, dijelaskan bahwa pengembangan sistem informasi yang baik melalui *SDLC* dapat membantu perusahaan dalam mengelola data akuntansi secara lebih efektif (Romney & Steinbart, 2018, hal. 50). Selain itu, buku "*Systems Analysis and Design*" oleh Kendall dan Kendall juga menekankan pentingnya *SDLC* dalam mengoptimalkan proses bisnis dan meningkatkan efisiensi operasional (Kendall & Kendall, 2019, hal. 72).

Melalui penelitian ini, akan dibahas bagaimana optimasi proses bisnis akuntansi dapat dicapai melalui tahapan *SDLC* yang efisien (Romney & Steinbart, 2018, hal. 52). Setiap tahap *SDLC* akan dianalisis untuk mengidentifikasi langkah-langkah yang dapat meningkatkan efisiensi dan efektivitas sistem akuntansi, mulai dari perencanaan hingga pemeliharaan (Kendall & Kendall, 2019, hal. 75). Dengan demikian, diharapkan penelitian ini dapat memberikan panduan praktis bagi perusahaan dalam mengoptimalkan proses bisnis akuntansi mereka (Romney & Steinbart, 2018, hal. 55).

Berkembangnya kebutuhan perangkat lunak membuat perancangan sebuah sistem menjadi sebuah kebutuhan pula, terutama untuk sistem yang besar, sulit dan memiliki fitur kompleks. Berbagai metodologi pun mulai digunakan untuk memudahkan dalam pengembangan sistemnya (Nukman et al., 2014).

Terdapat beberapa metode yang dapat kita gunakan dalam pengembangan sistem pada jurnal ini menggunakan metode *Waterfall*. Metode ini memiliki kelebihan dan kekurangan, untuk digunakan dalam mengembangkan sebuah sistem informasi (Nukman et al., 2014).

Kekurangan menggunakan metode *Waterfall* adalah metode ini tidak memungkinkan untuk banyak revisi jika terjadi kesalahan dalam prosesnya. Karena setelah aplikasi ini dalam tahap pengujian, sulit untuk kembali lagi dan mengubah sesuatu yang tidak terdokumentasi dengan baik dalam tahap konsep sebelumnya. (Binus 2019) pada jurnal (Mochammad Haldi Widiyanto 2019).

LANDASAN TEORI

Optimasi Proses Bisnis Akuntansi

Optimasi proses bisnis akuntansi merupakan langkah penting untuk meningkatkan efisiensi dan efektivitas operasional dalam organisasi. Proses ini melibatkan analisis dan perbaikan berbagai tahap dalam siklus akuntansi untuk meminimalkan kesalahan,

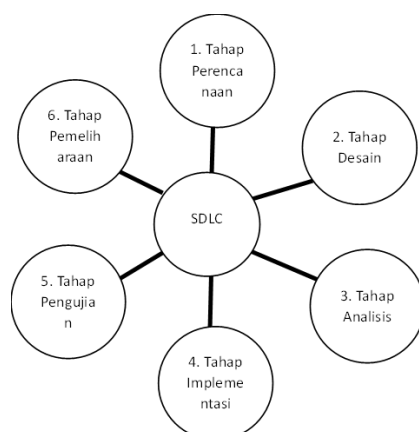
mempercepat waktu proses, dan mengurangi biaya (Romney & Steinbart, 2018, hal. 55). Menurut Romney dan Steinbart (2018, hal. 60), penerapan sistem informasi akuntansi yang efektif dapat meningkatkan kualitas data yang dihasilkan, sehingga mempermudah pengambilan keputusan yang lebih baik.

System Development Lifecycle (SDLC)

SDLC adalah kerangka kerja yang digunakan untuk mengembangkan dan memelihara sistem informasi, termasuk sistem akuntansi. *SDLC* terdiri dari beberapa tahapan utama, yaitu perencanaan, analisis, desain, implementasi, pengujian, dan pemeliharaan (Kendall & Kendall, 2019, hal. 67). Setiap tahap dalam *SDLC* dirancang untuk memastikan bahwa sistem yang dikembangkan memenuhi kebutuhan bisnis dan dapat diimplementasikan secara efisien.

METODOLOGI PENELITIAN

Metode yang dipakai didalam artikel jurnal ini adalah *System Development Life Cycles SDLC*. Tahapan dari metode *SDLC* ini adalah sebagai berikut :



Gambar 1. Tahapan Pembangunan Sistem Informasi dengan metode *SDLC*

1) Tahap Perencanaan: Perencanaan merupakan tahap awal yang kritis dalam *SDLC*. Pada tahap ini, tujuan proyek didefinisikan, dan sumber daya yang diperlukan diidentifikasi (Kendall & Kendall, 2019, hal. 70). Menurut Kendall dan Kendall (2019, hal. 72), perencanaan yang baik dapat membantu mengidentifikasi potensi risiko dan menyediakan kerangka kerja untuk manajemen proyek yang lebih baik; 2) Tahap Analisis: Tahap analisis melibatkan pengumpulan dan evaluasi kebutuhan sistem. Analisis yang menyeluruh memungkinkan pengembangan sistem yang sesuai dengan kebutuhan bisnis dan dapat meningkatkan efisiensi proses akuntansi (Romney & Steinbart, 2018, hal. 110). Kendall dan Kendall (2019, hal. 75) menekankan pentingnya melibatkan pengguna akhir dalam tahap ini untuk memastikan semua kebutuhan teridentifikasi dengan baik; 3) Tahap Desain: Desain sistem melibatkan pembuatan blueprint atau rancangan sistem yang akan dikembangkan. Desain yang baik memastikan

bahwa semua kebutuhan bisnis terpenuhi dan sistem dapat diimplementasikan dengan efektif (Kendall & Kendall, 2019, hal. 115). Menurut Pressman (2014, hal. 100), desain yang baik juga harus mempertimbangkan aspek keamanan dan skalabilitas sistem; 4) Tahap Implementasi: Implementasi adalah tahap di mana sistem yang dirancang diimplementasikan dalam lingkungan operasional. Proses ini mencakup instalasi perangkat keras dan perangkat lunak serta pelatihan pengguna (Romney & Steinbart, 2018, hal. 140). Kendall dan Kendall (2019, hal. 130) mencatat bahwa komunikasi yang efektif antara tim pengembang dan pengguna akhir sangat penting selama tahap ini untuk memastikan bahwa sistem diimplementasikan sesuai dengan desain; 5. Tahap Pengujian: Pengujian memastikan bahwa sistem berfungsi sebagaimana mestinya dan memenuhi semua kebutuhan yang telah diidentifikasi pada tahap analisis. Pengujian yang menyeluruh dapat mengidentifikasi dan memperbaiki kesalahan sebelum sistem digunakan secara penuh (Pressman, 2014, hal. 120). Menurut Kendall dan Kendall (2019, hal. 135), pengujian yang komprehensif mencakup berbagai jenis pengujian, termasuk pengujian fungsional, pengujian kinerja, dan pengujian keamanan; 6. Tahap Pemeliharaan: Pemeliharaan adalah tahap di mana sistem yang telah diimplementasikan dipantau dan diperbaiki sesuai kebutuhan. Pemeliharaan yang efektif memastikan bahwa sistem tetap relevan dan efisien dalam jangka panjang (Kendall & Kendall, 2019, hal. 150). Romney dan Steinbart (2018, hal. 160) menekankan bahwa pemeliharaan juga mencakup pembaruan sistem untuk menyesuaikan dengan perubahan dalam lingkungan bisnis dan teknologi.

HASIL DAN PEMBAHASAN

SDLC (*Systems Development Life Cycle*, Siklus Hidup Pengembangan Sistem) atau *Systems Life Cycle* (Siklus Hidup Sistem), dalam rekayasa sistem dan rekayasa perangkat lunak, adalah proses pembuatan dan perubahan sistem serta model dan metodologi yang digunakan untuk mengembangkan sistem-sistem tersebut. Konsep ini umumnya merujuk pada sistem komputer atau informasi (Pratama & Sedyono, 2017), Dalam rekayasa perangkat lunak, konsep *SDLC* mendasari berbagai jenis metodologi pengembangan perangkat lunak.

Analisis Implementasi *SDLC* dalam Proses Bisnis Akuntansi

Implementasi *System Development Lifecycle* (*SDLC*) dalam proses bisnis akuntansi merupakan pendekatan yang sangat strategis untuk meningkatkan efisiensi dan efektivitas operasional. Setiap tahap dalam *SDLC* memiliki peran penting dalam memastikan bahwa sistem informasi akuntansi yang dikembangkan memenuhi kebutuhan bisnis dan dapat dioperasikan secara efisien. Kendall dan Kendall (2019, hal. 67) mengemukakan bahwa *SDLC*

terdiri dari beberapa tahapan utama: perencanaan, analisis, desain, implementasi, pengujian, dan pemeliharaan, yang semuanya harus dilaksanakan dengan cermat untuk mencapai hasil yang optimal.

Tahap Perencanaan

Tahap perencanaan adalah langkah awal dalam *SDLC* yang berfokus pada identifikasi kebutuhan proyek dan perumusan rencana yang rinci. Dalam konteks akuntansi, tahap ini sangat penting karena membantu mengidentifikasi kebutuhan spesifik sistem akuntansi, termasuk kebutuhan pelaporan keuangan dan kepatuhan terhadap peraturan. Menurut Kendall dan Kendall (2019, hal. 70), perencanaan yang baik dapat mengurangi risiko proyek dan memastikan bahwa proyek berjalan sesuai dengan jadwal dan anggaran yang telah ditetapkan.

Selama tahap perencanaan, tim proyek harus melakukan analisis mendalam terhadap tujuan bisnis, mengidentifikasi pemangku kepentingan, dan menentukan ruang lingkup proyek. Romney dan Steinbart (2018, hal. 70) menekankan bahwa perencanaan yang efektif juga harus mencakup identifikasi potensi kendala dan risiko serta penyusunan strategi mitigasi yang sesuai. Selain itu, perencanaan yang baik membantu memastikan bahwa semua sumber daya yang diperlukan, termasuk anggaran, waktu, dan tenaga kerja, telah dipertimbangkan dan dialokasikan dengan tepat.

Tahap Analisis

Tahap analisis dalam *SDLC* melibatkan pengumpulan dan evaluasi kebutuhan sistem secara mendetail. Dalam proses bisnis akuntansi, tahap ini melibatkan analisis kebutuhan pengguna akhir, termasuk kebutuhan pelaporan dan analisis data keuangan. Romney dan Steinbart (2018, hal. 110) menyatakan bahwa analisis yang menyeluruh memungkinkan pengembangan sistem yang sesuai dengan kebutuhan bisnis, yang pada gilirannya dapat meningkatkan efisiensi proses akuntansi.

Menurut Kendall dan Kendall (2019, hal. 75), melibatkan pengguna akhir dalam tahap ini sangat penting untuk memastikan bahwa semua kebutuhan diidentifikasi dengan tepat. Pendekatan ini membantu dalam merumuskan spesifikasi sistem yang jelas dan komprehensif, yang akan menjadi dasar bagi tahap desain. Analisis yang tepat juga membantu mengidentifikasi proses bisnis yang dapat diotomatisasi atau disederhanakan, yang dapat mengurangi beban kerja manual dan meningkatkan akurasi data.

Tahap Desain

Tahap desain melibatkan pembuatan blueprint atau rancangan sistem yang akan dikembangkan. Dalam konteks sistem akuntansi, desain harus mencakup struktur data, aliran informasi, dan antarmuka pengguna yang intuitif. Kendall dan Kendall (2019, hal. 115)

mengemukakan bahwa desain yang baik memastikan bahwa semua kebutuhan bisnis dapat diakomodasi dengan baik dan sistem dapat diimplementasikan dengan efektif.

Desain yang baik harus mempertimbangkan berbagai aspek, termasuk keamanan, skalabilitas, dan kemudahan penggunaan. Pressman (2014, hal. 100) menyatakan bahwa desain yang baik juga harus mencakup rencana untuk pengujian dan pemeliharaan sistem. Dalam tahap ini, tim pengembang harus bekerja sama dengan pemangku kepentingan untuk memastikan bahwa desain yang dihasilkan sesuai dengan kebutuhan dan harapan mereka. Prototyping dapat digunakan dalam tahap ini untuk memberikan gambaran awal tentang bagaimana sistem akan berfungsi, yang dapat membantu mengidentifikasi dan mengatasi masalah potensial sebelum tahap implementasi dimulai.

Tahap Implementasi

Tahap implementasi adalah tahap di mana sistem yang telah dirancang diimplementasikan dalam lingkungan operasional. Implementasi mencakup instalasi perangkat keras dan perangkat lunak, konfigurasi sistem, dan pelatihan pengguna akhir. Romney dan Steinbart (2018, hal. 140) mencatat bahwa keberhasilan implementasi sangat bergantung pada komunikasi yang efektif antara tim pengembang dan pengguna akhir.

Kendall dan Kendall (2019, hal. 130) menekankan bahwa selama tahap ini, tim proyek harus memastikan bahwa semua komponen sistem diinstal dan dikonfigurasi dengan benar. Pelatihan pengguna juga merupakan bagian penting dari tahap implementasi, karena memastikan bahwa pengguna akhir dapat menggunakan sistem baru dengan efektif. Pelatihan harus mencakup penggunaan dasar sistem, serta pemecahan masalah umum dan prosedur untuk melaporkan masalah yang lebih kompleks.

Tahap Pengujian

Tahap pengujian dalam *SDLC* memastikan bahwa sistem berfungsi sebagaimana mestinya dan memenuhi semua kebutuhan yang telah diidentifikasi pada tahap analisis. Pengujian dalam sistem akuntansi melibatkan verifikasi fungsi-fungsi utama seperti pelaporan keuangan, pengelolaan buku besar, dan rekonsiliasi. Pressman (2014, hal. 120) menyatakan bahwa pengujian yang menyeluruh dapat mengidentifikasi dan memperbaiki kesalahan sebelum sistem digunakan secara penuh.

Pengujian yang komprehensif mencakup berbagai jenis pengujian, termasuk pengujian fungsional, pengujian kinerja, dan pengujian keamanan. Kendall dan Kendall (2019, hal. 135) menekankan bahwa pengujian fungsional memastikan bahwa setiap fitur sistem bekerja sesuai dengan spesifikasi, sedangkan pengujian kinerja memastikan bahwa sistem dapat menangani

beban kerja yang diharapkan. Pengujian keamanan penting untuk memastikan bahwa data keuangan yang sensitif dilindungi dari ancaman keamanan.

Tahap Pemeliharaan

Tahap pemeliharaan melibatkan pemantauan dan perbaikan sistem yang telah diimplementasikan untuk memastikan bahwa sistem tetap relevan dan efisien dalam jangka panjang. Pemeliharaan yang efektif mencakup pembaruan sistem untuk menyesuaikan dengan perubahan dalam lingkungan bisnis dan teknologi, serta perbaikan bug dan peningkatan kinerja. Kendall dan Kendall (2019, hal. 150) menyatakan bahwa pemeliharaan yang baik memastikan bahwa sistem terus memenuhi kebutuhan bisnis dan pengguna akhir.

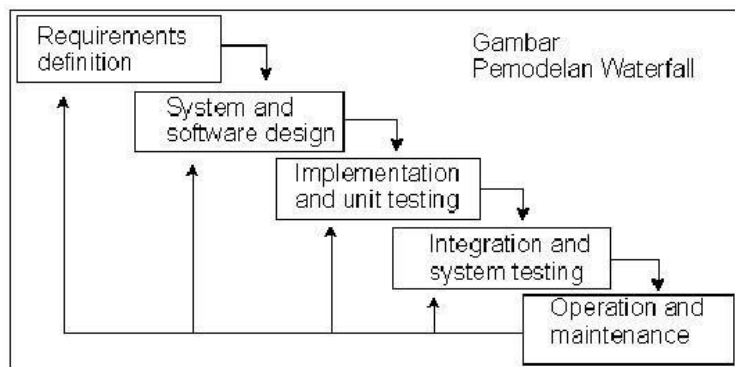
Romney dan Steinbart (2018, hal. 160) menekankan bahwa pemeliharaan juga mencakup respons terhadap umpan balik pengguna untuk terus meningkatkan kinerja sistem. Tim pemeliharaan harus siap untuk menangani masalah yang muncul dan melakukan pembaruan sistem secara berkala untuk memastikan bahwa sistem tetap aman dan dapat diandalkan. Pemeliharaan yang proaktif dapat mencegah masalah besar sebelum mereka mempengaruhi operasi bisnis.

Model SDLC untuk Proses Bisnis Akuntansi

Model *Waterfall*

Model *Waterfall* adalah model pengembangan perangkat lunak yang paling sering digunakan dalam pengembangan sebuah sistem. Menurut Sommerville, model pengembangan ini bersifat linear dari tahap awal pengembangan sistem yaitu tahap perencanaan sampai tahap akhir pengembangan sistem yaitu tahap pemeliharaan. Tahapan berikutnya tidak akan dilaksanakan sebelum tahapan sebelumnya selesai dilaksanakan dan tidak bisa kembali atau mengulang ke tahap sebelumnya (Wahid, 2020).

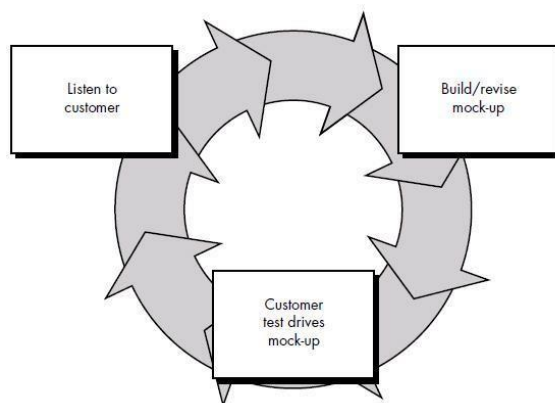
Tahapan dari model ini dapat dilihat pada gambar dibawah ini.



Gambar 1. Model Pengembangan Waterfall

Model *Prototype*

Model *Prototype* merupakan suatu teknik untuk mengumpulkan informasi tertentu mengenai kebutuhan-kebutuhan informasi pengguna secara cepat. Berfokus pada penyajian dari aspek-aspek perangkat lunak tersebut yang akan nampak bagi pelanggan atau pemakai. Sistem dengan model *Prototype* memungkinkan pengguna agar mengetahui seperti apa tahapan sistem dibuat sehingga sistem mampu beroperasi dengan baik (yoko Et al.,2019) *Prototype* tersebut akan dievaluasi oleh pelanggan/pemakai dan dipakai untuk menyaring kebutuhan pengembangan perangkat lunak. Dengan metode *Prototype* penerapan sistem menjadi semakin mudah karena adanya peran aktif dari pengguna. Namun metode ini dipergunakan dengan tujuan ingin mengambil langsung masukan dari penggunanya. Dan juga mempermudah pengguna agar lebih akrab dengan sistem yang akan dikembangkan. Tahapan dari model ini dapat dilihat pada gambar dibawah ini.



Gambar 2. Model *Prototype* menurut Roger S. Pressman, Ph.D.

KESIMPULAN DAN SARAN

Optimasi proses bisnis akuntansi melalui tahapan *System Development Lifecycle* (*SDLC*) yang efisien dapat memberikan manfaat signifikan dalam meningkatkan efisiensi dan efektivitas operasional sistem akuntansi. Dengan menerapkan setiap tahap *SDLC* secara cermat—mulai dari perencanaan, analisis, desain, implementasi, pengujian, hingga pemeliharaan—perusahaan dapat memastikan bahwa sistem informasi akuntansi yang dikembangkan memenuhi kebutuhan bisnis secara komprehensif dan dapat dioperasikan dengan efisien (Kendall & Kendall, 2019, hal. 67).

Tahap perencanaan membantu mengidentifikasi kebutuhan spesifik dan potensi risiko proyek, yang sangat penting untuk memastikan keberhasilan proyek (Romney & Steinbart, 2018, hal. 70). Analisis kebutuhan yang mendalam memungkinkan pengembangan sistem yang sesuai dengan kebutuhan pengguna akhir, sementara desain yang baik memastikan bahwa

semua kebutuhan bisnis diakomodasi dan sistem dapat diimplementasikan dengan efektif (Kendall & Kendall, 2019, hal. 115).

Implementasi yang sukses sangat bergantung pada komunikasi yang efektif antara tim pengembang dan pengguna akhir, serta pelatihan yang memadai untuk memastikan pengguna dapat menggunakan sistem baru dengan efektif (Romney & Steinbart, 2018, hal. 140). Pengujian yang komprehensif memastikan bahwa sistem berfungsi sesuai spesifikasi dan dapat diandalkan dalam operasional sehari-hari (Pressman, 2014, hal. 120). Tahap pemeliharaan yang proaktif memastikan bahwa sistem tetap relevan dan efisien dalam jangka panjang serta dapat beradaptasi dengan perubahan kebutuhan bisnis (Kendall & Kendall, 2019, hal. 150).

Pemilihan model *SDLC* yang tepat berdasarkan kebutuhan spesifik proyek dan lingkungan bisnis sangat penting. Model *Waterfall*, *Spiral*, *Rapid Application Development (RAD)*, dan *Prototyping* masing-masing memiliki kelebihan dan kekurangan yang harus dipertimbangkan dengan hati-hati untuk mencapai hasil yang optimal (Pressman, 2014, hal. 39; Boehm, 1988, hal. 64; Martin, 1991, hal. 28; Kendall & Kendall, 2019, hal. 93).

Dengan menerapkan pendekatan *SDLC* yang terstruktur dan fleksibel, perusahaan dapat memastikan bahwa sistem akuntansi mereka tidak hanya memenuhi kebutuhan saat ini tetapi juga siap untuk beradaptasi dengan perubahan di masa depan. Hal ini akan membantu perusahaan dalam menjaga efisiensi operasional dan mempertahankan keunggulan kompetitif di pasar yang terus berkembang (Kendall & Kendall, 2019, hal. 67; Romney & Steinbart, 2018, hal. 55).

DAFTAR REFERENSI

- Boehm, B. W. (1988). A spiral model of software development and enhancement. *ACM SIGSOFT Software Engineering Notes*, 11(4), 14-24.
- Eureka Media Aksara. (2023). *Memahami System Development Life Cycle*.
- Hanif, R., & Yuki, F. (2018). Pengembangan sistem informasi aset tetap pemerintah kabupaten x: System Development Life Cycle (SDLC) teknik Waterfall.
- Kendall, K. E., & Kendall, J. E. (2019). *Systems Analysis and Design*.
- Martin, J. (1991). *Rapid Application Development*. Macmillan Publishing Company.
- Murdiani, D., & Muhamad, S. (2022). Perbandingan Metodologi Waterfall dan RAD (Rapid Application Development) dalam Pengembangan Sistem Informasi. *Jurnal Informatika Teknologi dan Sains*.
- Nilma, (2022). *Jurnal Publikasi Teknik Informatika "Penerapan Model Software Development Life Cycle Pada Rancang Bangun Sistem Payroll Perusahaan"*.

- Nilma. (2022). Penerapan Model Software Development Life Cycle pada rancang bangun sistem payroll perusahaan. *Jurnal Publikasi Teknik Informatika*. Raihan Hanif, Firmanto Yuki, (2018). Pengembangan system informasi asset tetap pemerintah kabupaten x : System Developmen Life Cycle (SDLC) Teknik Waterfal
- Ningsih, W., & Habibah, N. (2023). Perbandingan model Waterfall dan metode Prototype untuk pengembangan aplikasi pada sistem informasi.
- Pressman, R. S. (2014). *Software Engineering: A Practitioner's Approach*.