

Article history

Received April 25, 2022

Accepted January 16, 2023

**STUDI LITERATUR EFISIENSI MODEL RAPID APPLICATION DEVELOPMENT
DALAM PENGEMBANGAN PERANGKAT LUNAK (2014-2022)****Rizky Parluka, Muhammad Affudin, Ilham Akbar Pradana, Yohanes Dimas Wisnu Wiratama,
Mohammad Nur Holis**

Fakultas Ilmu Komputer, Universitas Pembangunan Nasional “Veteran” Jawa Timur

Email: rizkyparlika.if@upnjatim.ac.id, muhafifudin27@gmail.com, ilhamakbar.pradana@gmail.com,
20081010112@student.upnjatim.ac.id, cholissanctis@gmail.com**Abstract**

Software Development Life Cycle (SDLC) or software development cycle is defined as systematic stages in order to produce good and quality software and can meet customer desires and can be used to meet needs and be used by users. SDLC itself has several general stages, namely 1) Requirements Analysis Phase, 2) Design Phase, 3) Implementation and coding phase, 4) Validation and testing phase, 6) Maintenance phase. Rapid Application Development or RAD is one of the SDLC methodologies that is widely used for development teams that prioritize time efficiency in software development. Because this methodology is considered a methodology that has the shortest processing time. This study aims to analyze the efficiency level of the RAD model in the world of software development. The author uses the narrative review method, which will be applied to the previously collected literature. In this study, it will be concluded the things that make the RAD methodology the most efficient model, supporting and inhibiting the RAD model, as well as comparing the RAD methodology with other development models.

Keywords: *Software Development Life Cycle (SDLC), Rapid Application Development (RAD), Efficiency*

Abstrak

*Software Development Life Cycle (SDLC) atau siklus pengembangan perangkat lunak didefinisikan sebagai tahapan-tahapan sistematis guna menghasilkan perangkat lunak yang baik dan berkualitas serta dapat memenuhi keinginan customer dan dapat digunakan untuk memenuhi kebutuhan dan digunakan oleh user. SDLC sendiri memiliki beberapa tahapan umum, yaitu 1) Fase Analisis Kebutuhan, 2) Fase Desain, 3) Fase implementasi dan pengkodean, 4) Fase validasi dan pengujian, 6) Fase Pemeliharaan. Rapid Application Development atau RAD merupakan salah satu metodologi SDLC yang marak digunakan untuk tim pengembang yang mengutamakan efisiensi waktu dalam pengembangan perangkat lunak. Karena metodologi ini dinilai merupakan metodologi yang memiliki waktu pengerjaan paling singkat. Penelitian ini memiliki tujuan guna menganalisis tingkat efisiensi dari model RAD dalam dunia pengembangan perangkat lunak. Penulis menggunakan metode *narrative review*, yang akan diterapkan terhadap literatur-literatur yang telah dikumpulkan sebelumnya. Dalam penelitian ini akan disimpulkan hal-hal yang membuat metodologi RAD dinilai sebagai model yang paling efisien, penunjang dan penghambat model RAD, juga perbandingan metodologi RAD dengan model pengembangan lainnya.*

Kata Kunci: *Software Development Life Cycle (SDLC), Rapid Application Development (RAD), Efisiensi*

1. PENDAHULUAN

Latar Belakang

Dewasa ini, perkembangan teknologi perangkat lunak dapat dikatakan telah menjangkiti segala bidang seperti kesehatan, pendidikan, bisnis, hingga politik pemerintahan. Banyak kemudahan yang didapatkan dari makin berkembangnya teknologi ini, terutama dari segi efisiensi, ketepatan, dan kebaruan informasi yang dihasilkan dari berbagai perangkat lunak yang tersedia saat ini. Bersamaan dengan semakin berkembangnya teknologi perangkat lunak, ikut berkembang pula berbagai macam metodologi untuk mengembangkannya.

Namun jika kita telaah lebih teliti, dari sekian banyaknya proyek pembangunan perangkat lunak, kendala yang sering ditemui yaitu kendala waktu dan biaya [1]. Untuk itu dalam pengembangan perangkat lunak, diperlukan sebuah model untuk proses pengembangannya. Tujuan digunakannya model pengembangan perangkat lunak ini adalah untuk menyediakan alur atau langkah-langkah dalam pengembangan perangkat lunak. Dengan adanya alur pengerjaan yang jelas, produk yang dihasilkan dari proyek pengembangan perangkat lunak diharapkan memiliki kualitas yang tinggi dengan waktu pengembangan yang singkat, serta biaya yang lebih rendah. Bertepatan dengan semakin berkembangnya teknologi, makin bermunculan pula berbagai model yang dapat dipilih untuk digunakan sebagai metode mengembangkan perangkat lunak. Salah satu model pengembangan perangkat lunak yang banyak digunakan dan dianggap lebih efisien daripada model yang lainnya adalah RAD.

Rapid Application Development (RAD) merupakan sebuah metodologi untuk mengembangkan perangkat lunak yang hanya memerlukan perencanaan minimal untuk rapid prototyping [2]. Dari definisi konsep RAD ini, dapat diartikan bahwa pengimplementasian RAD bertujuan untuk mempercepat proses pengembangan perangkat lunak dan diekspektasikan dapat menghasilkan produk yang lebih berkualitas. Dapat dikatakan bahwa model RAD adalah adaptasi dari model tradisional (waterfall) yang lebih cepat dan efektif [3]. *Rapid Application Development* lebih menitikberatkan kegiatan tahap pengembangan daripada tahap perencanaan. Metodologi *Rapid Application Development* ini cocok digunakan untuk proyek yang dapat “dipisah-pisahkan” dalam beberapa

modul baik proyek skala kecil, menengah, maupun besar [4].

Studi analisis ini bertujuan untuk mengetahui tingkat keefektifan model RAD jika dibandingkan dengan model pengembangan lain. Dengan acuan dari penelitian dan jurnal-jurnal yang telah dikumpulkan, akan diteliti apakah model RAD dapat dinilai lebih efisien dari segi waktu dan segi sumber daya dibandingkan model yang lain dalam pengembangan sebuah perangkat lunak.

Rumusan Masalah

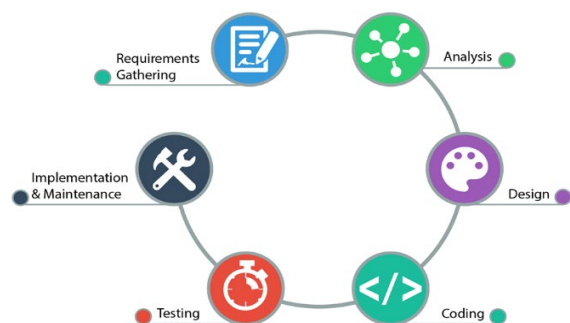
1. Bagaimana efisiensi dari pengembangan perangkat lunak dengan model *Rapid Application Development* dibandingkan model tradisional.
2. Bagaimana aspek penunjang dan penghambat mempengaruhi model *Rapid Application Development* dalam pengembangan perangkat lunak.

Tujuan Penelitian

1. Untuk mengetahui efisiensi dari pengembangan perangkat lunak dengan model *Rapid Application Development* dibandingkan model tradisional.
2. Untuk mengetahui aspek penunjang dan penghambat mempengaruhi model *Rapid Application Development* dalam pengembangan perangkat lunak.

Software Development Life Cycle (SDLC)

Siklus Daur hidup pengembangan perangkat lunak atau *Software Development Life Cycle* atau lebih dikenal dengan sebutan SDLC merupakan model atau metode pengembangan perangkat lunak yang digunakan pada saat sebuah perangkat lunak pertama kali akan dikembangkan [5]. Tahapan paling dasar dalam SDLC adalah sebagai berikut:



Gambar 1. Ilustrasi Tahapan SDLC. Sumber: <http://mcastud.com/student-project-development-go/>

1. Tahap perencanaan (*planning*), dalam tahap ini didefinisikan perkiraan kebutuhan apa saja yang akan dibutuhkan oleh perangkat lunak.
2. Tahap analisis, dalam tahap ini dilakukan analisis terhadap perencanaan yang telah dibuat.
3. Tahap desain, dalam tahap ini akan ditentukan proses dan data yang akan digunakan dalam sistem. Akan dibuat rancangan mengenai bagaimana data akan diolah dalam sistem, bagaimana tampilan antar muka sistem, bagaimana sistem alur kerja sistem, dan semacamnya.
4. Tahap implementasi, desain yang dibuat akan diimplementasikan dalam bentuk kode.
5. Tahap pengujian (*testing*), setelah kode dapat dieksekusi menjadi sebuah program, akan dilakukan pengujian guna mengetahui segala kemungkinan cacat dan kesalahan dalam sistem.
6. Tahap pemeliharaan (*maintenance*). Dilakukan untuk menjaga sistem tetap dapat berjalan dengan baik dan tetap bisa memenuhi kebutuhan *customer*.

Dalam pengembangan perangkat lunak diperlukan beberapa peralatan pendukung untuk merancang sistem desain diantaranya:

1. Use Case

Use case merupakan bentuk pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat [6]. Diagram ini sering digunakan dalam pengembangan perangkat lunak untuk mengetahui kebutuhan fungsional dari perangkat lunak yang dibangun. Diagram ini nantinya akan menggambarkan hubungan/interaksi yang akan dilakukan oleh user dan sistem yang dirancang. Use case diagram akan membantu tim pengembang dalam menganalisa kebutuhan yang sesuai untuk user juga menggambarkan tahapan-tahapan dalam proses bisnis (*business process*) secara transparan untuk menanggulangi adanya kesalahan pada perangkat lunak yang dikembangkan.

2. Activity Diagram

Diagram ini akan mempermudah tim pengembang dalam mensketsakan alur kerja dari perangkat lunak yang tengah dikembangkan.

3. Entity Relationship Diagram (ERD)

Diagram ini digunakan untuk mempermudah tim pengembang dalam memodelkan basis data. Didalamnya akan digambarkan relasi antara entitas yang terdapat dalam basis data dari perangkat lunak yang dibangun.

4. Logical Record Structure (LRS)

Melalui proses kardinalitas, ERD yang telah dibuat akan ditransformasikan menjadi LRS.

Efisiensi

Menurut KBBI, efisiensi adalah kemampuan menjalankan tugas dengan baik [7]. Istilah efisiensi adalah istilah yang sering dijumpai dalam pekerjaan atau kegiatan-kegiatan industri. Efisiensi dapat diartikan sebagai upaya penggunaan input dengan menggunakan sumber daya seminimal mungkin untuk menghasilkan output atau hasil yang semaksimal mungkin. Sumber daya yang dimaksud dalam kegiatan efisiensi tersebut adalah tenaga, uang, dan waktu [8]. Dengan diterapkannya efisiensi diharapkan bahwa dalam suatu kegiatan produksi tidak akan terjadi pemborosan sumber daya yang telah disebutkan.

Mengingat definisi dari efisiensi, konsep efisiensi tidak hanya dapat diterapkan dalam kegiatan industri, melainkan juga diterapkan dalam kehidupan sehari-hari. Untuk itu, manusia akan terpicu untuk menemukan penemuan-penemuan untuk membantu kegiatan dan pekerjaan sehari-hari. Penemuan-penemuan ini dapat berupa alat maupun model-model untuk membantu kegiatan dan pekerjaan sehari-harinya.

Hubungan konsep efisiensi dengan pengembangan perangkat lunak adalah dalam model pengembangan perangkat lunak. Model-model pengembangan tersebut digunakan untuk menghindari pemborosan sumber daya dan meminimalisir hambatan-hambatan yang mungkin akan terjadi di tengah pengembangan perangkat lunak. Dalam mencapai keadaan efisien dapat membutuhkan langkah-langkah atau skenario yang berbeda [8]. Oleh karena itu dikembangkanlah berbagai model pengembangan perangkat lunak dengan pendekatan yang berbeda-beda pula.

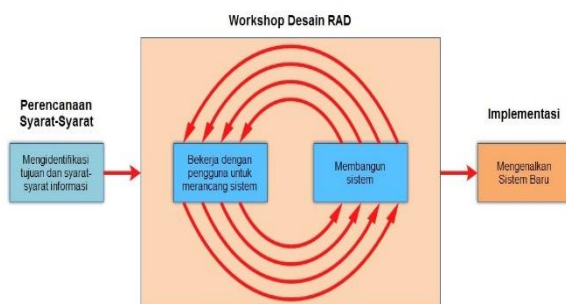
Rapid Application Development (RAD)

Model RAD merupakan salah satu dari sekian banyak model pengembangan perangkat lunak yang ada. RAD memiliki ciri khas yaitu pengembangan perangkat lunak yang menggunakan model ini akan memakan waktu yang relatif singkat. Untuk pengembangan suatu perangkat lunak yang umumnya memakan waktu minimal 180 hari. Namun dengan model RAD suatu sistem dapat diselesaikan hanya dalam waktu 60-90 hari [9]. Model ini cocok digunakan saat alokasi waktu yang digunakan untuk

pengembangan aplikasi memiliki batasan yang pendek. Waktu pada tahapan perencanaan hanya dialokasikan secara singkat karena proses pengembangan aplikasi akan berjalan secara fleksibel sesuai kebutuhan.

Berikut tahapan-tahapan dalam pelaksanaan model RAD :

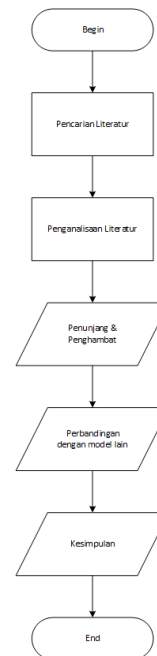
1. *Requirement Planning* (Perencanaan Syarat-syarat)
 Pada fase ini, system analyst, customer dan user akan bertemu dan Bersama-sama merumuskan tujuan serta requirements dari perangkat lunak yang akan dikembangkan.
2. *RAD Design Workshop* (Workshop Design RAD)
 Pada fase ini akan dibuat sketsa dan diagram yang nantinya akan dijadikan panduan dalam pengembangan perangkat lunak. ERD, UML, dan diagram-diagram lain akan dirancang oleh system analyst Bersama dengan programmer pada tahap ini. Disini juga akan dibuat pula rancangan dari tampilan antarmuka dari perangkat lunak yang dibuat.
3. *Implementation* (Implementasi)
 Kode dari perangkat lunak akan dibuat pada fase ini. Akan pula dilakukan pengujian oleh QA dan oleh user untuk memutuskan apakah sistem sudah layak untuk digunakan. Fase ini adalah tahapan memperkenalkan sistem kepada *user interface* [10]. Dalam fase ini, terutama dalam tahap pengujian, user akan semakin sering dilibatkan dalam proses pengembangan perangkat lunak. Tahap ini bisa dikatakan sebagai tahapan terakhir dalam model RAD, karena jika perangkat lunak telah dinilai layak oleh user dan QA dari proses pengujian, maka perangkat lunak dapat dirilis dan digunakan.



Gambar 2. Workshop RAD. Sumber: <https://piyaneo.files.wordpress.com/2014/05/siklus-rad-e1399723570398.jpg>

2. METODE PENELITIAN

Tulisan ini merupakan studi literatur, jadi tidak seperti jurnal penelitian lain, studi literatur tidak akan menyajikan data baru, namun akan menyajikan data-data dari penelitian sebelumnya untuk dianalisa [11]. Penelitian ini menggunakan model *Narrative Literature Review* (NLR) yang akan diterapkan pada literatur-literatur yang telah dikumpulkan. NLR bertujuan untuk mengidentifikasi dan meringkas literatur-literatur yang telah diterbitkan sebelumnya. Literatur ditelusuri melalui Google Scholar, Core.uk, dan Science and Technology Index (Sinta). Adapun kriteria literatur yang akan diteliti dalam penelitian ini berupa artikel jurnal yang diterbitkan dalam kurun waktu 2013 - 2022. Periode waktu ini dipilih karena dinilai menjadi periode yang cocok untuk dianalisis karena tidak terlalu lama, dan tidak juga penelitian yang dilakukan pun bisa dikatakan tidak sedikit. Hingga hasil dari penelitian ini nantinya bisa dikatakan valid.



Gambar 3. Flowchart Metode NLR

Kata kunci pencarian yakni “pengembangan perangkat lunak RAD”, “model *Rapid application Development*”, “*software development*”, “*RAD software Development*”. Semua artikel dan jurnal yang relevan kemudian dipertimbangkan menurut kriteria sebagai syarat untuk dapat digunakan sebagai data studi literatur. Berikut data literatur yang dianalisis.

Tabel 1. Daftar jurnal yang dianalisis

No	Penulis, Tahun	Luaran
1	Sandy Kosasi (2015)	Sistem Perniagaan Elektronik Furniture.
2	Ni Wayan S. S., Ni W. Wardani, Ketut L. Maswari, & I Dewa M. K. Muku (2021)	Sistem Informasi Aplikasi Payroll STMIK STIKOM berbasis Web.
3	Wahyu Wijaya Widiyanto (2018)	Sistem informasi kepegawaian (SIMPEG) .
4	Rosa Delima, Halim Budi Santosa, & Joko Purwadi (2017)	Dutanani Website, bagian dari Integrated Agriculture Information System (IAIS).
5	Nur Nafara Rofiq & Siti Muniroh (2018)	Aplikasi Pendaftaran Rawat Jalan Berbasis Web Rumah Sakit Bros Tangerang.
6	Suyanto & Andri (2020)	Aplikasi Pelaporan Kerusakan Jalan berbasis web dan mobile Dinas Bina Marga dan Tata Ruang Sumsel.
7	Yahya Dwi Wijaya (2020)	Sistem Informasi Toko Berkah.
8	AD. Supriatna (2018)	Sistem Informasi Perpustakaan.
9	Anik Adriani & Esti Qurniati (2018)	Sistem Informasi Toko Online.
10	A. K. Nalendra (2020)	Sistem Irigasi Agrikultur berbasis Internet Of Things (IoT).
11	Ahmad Chusyairi, Bintang Rudiawan & Eko Sasongko (2018)	Sistem Layanan Pelaporan Kehilangan (E-Report) Polres Banyuwangi
12	Jadiaman P., Ade Chandra S., & Caca handika (2022)	Aplikasi BK (Bimbingan Konseling) SMK Negeri 1 Muara Teweh.
13	Aris Baihaqi & Tumini (2021)	Sistem Aplikasi Pemesanan Menu Kedai kopi berbasis mobile.
14	Ade Suryanto & M. Ibnu Malik (2022)	Sistem Informasi pelayanan warga RT/RW 07/04 Kelurahan Jati.
15	Shoahudin al Ayubi & Yusnia Budiarta (2022)	Sistem informasi e-pegawai Yayasan Bina Insan kamil Jakarta
16	Mira Fauzia (2022)	Sistem pemesanan dan informasi jasa makeup berbasis mobile.

3. HASIL DAN PEMBAHASAN

Berdasarkan literatur-literatur yang telah dianalisis, telah terbukti bahwa model RAD Ketika digunakan dalam sebuah proyek pengembangan perangkat lunak dapat menghasilkan proyek yang baik dan memenuhi kebutuhan pengguna. Sandy Kosasi dalam penelitiannya menyatakan bahwa, melalui penerapan model RAD (*Rapid Application Development*) sudah dapat memberikan hasil maksimal dan secara signifikan sistem yang dikembangkan dapat memenuhi kebutuhan pengguna [12]. Model RAD dinilai tepat untuk menghasilkan sistem perangkat lunak yang diharapkan karena metodenya yang dinamis, fleksibel, dan melibatkan user secara langsung dalam tahap-tahap pengembangan. Lalu penelitian yang dijalankan oleh Ahmad Chusyairi, Bintang Rudiawan dan Eko Sasongko juga menunjukkan bahwa penggunaan model RAD dalam pengembangan Layanan Laporan Kehilangan (E-Report)) mempermudah dalam pembangunan layanan e-report berbasis web karena aktivitas dan fase yang dilalui banyak melibatkan user khususnya pada fase user design. Pengujian terhadap fitur pada layanan laporan kehilangan ini nantinya pada fase construction dan cutover dapat menampilkan sesuai dengan rencana kebutuhan yang ditetapkan oleh user [13]. A. D. Supriatna dalam penelitiannya dengan judul *Designing Library Information System using Rapud Application Development Method* juga menyatakan bahwa selain sistem yang dibuat dapat meningkatkan pelayanan terhadap member, implementasi pengembangan sistem informasi perpustakaan menggunakan model

Rapid Application Development dapat mengakselerasi proses dalam desain sistem [14].

Keberhasilan dari model RAD dalam pengembangan perangkat lunak juga dibuktikan oleh Anik Adriani dan Esti Qurniati dalam penelitian berjudul Sistem Informasi Penjualan Pada Toko Online Dengan Model *Rapid Application Development* (RAD) menyebutkan bahwa tahapan-tahapan pada RAD mempermudah pengembangan sistem karena tahapan-tahapannya yang sistematis dan cepat [15]. Suryanto dan Malik (2022) menyimpulkan bahwa sistem yang dibangun telah memenuhi kebutuhan tujuan penelitian tersebut. Diantaranya mempermudah dalam pengajuan surat pengantar, mempermudah warga melihat informasi kas RT, dan mempermudah ketua RT dalam membuat laporan [16]. Ayubi dan Budiarta (2022) menyatakan pernyataan bahwa sistem yang telah dirancang membuat administrasi izin dan cuti karyawan dapat terkontrol dengan baik serta dapat mengurangi biaya operasional dan memberikan kemudahan bagi karyawan [17]. Putra dan Lolly (2021) juga menyatakan bahwa sistem yang telah dibuat dengan model RAD telah mempermudah tahapan pencarian dan pencatatan data penjualan dengan data pemesanan dan data produk penjualan, seperti menambah, mengedit, dan menghapus data produk sehingga proses penjualan [18].

Wijaya (2020) menyatakan bahwa penerapan model RAD dinilai memiliki efisiensi waktu yang baik. Dapat dibuktikan juga dengan diterapkannya kerangka kerja yang terdiri dari 3 tahapan yaitu *planning*, *design*, dan *implementation* [19]. Tahapan-tahapan tersebut nantinya memiliki penjabaran tersendiri. Suryanto dan Andri (2020) menjabarkan tahap-tahap RAD dalam pengembangan aplikasi. Tahap pertama yaitu perencanaan, mengidentifikasi berbagai kebutuhan aplikasi yang akan dikembangkan. Tahap berikutnya yaitu desain. Dalam tahap ini diciptakan berbagai desain diagram untuk menggambarkan proses aplikasi diantaranya *use case diagram*, *activity diagram*, dan *class diagram*. Kemudian tahap yang terakhir yaitu tahap implementasi, tahap yang memaparkan bagaimana aplikasi bekerja dengan mengimplementasikan komponen-komponen yang ada dalam tahap sebelumnya [20].

Tujuan utama dari semua model pengembangan perangkat lunak adalah menghasilkan suatu perangkat lunak yang diharapkan memenuhi harapan dan kebutuhan para user. Namun, kejadian yang sering terjadi adalah tidak terlibatnya user dalam proses pengembangan perangkat lunak. Hal tersebut akan berakibat pada perangkat lunak yang dikembangkan jauh dari harapan user, sehingga meskipun tahap pengembangan telah usai dan perangkat lunak telah disetujui customer, namun user enggan menggunakannya. Lain halnya jika pada masa pengembangannya tim pengembang mengimplementasikan model RAD, dengan begitu user dapat menjadi bagian dalam tahap pengembangan perangkat lunak. RAD dapat menghasilkan suatu perangkat lunak dengan cepat karena perangkat lunak yang dikembangkan berhasil memenuhi harapan para pemakai yang akhirnya meminimalisir waktu yang dibutuhkan untuk pengembangan ulang setelah tahap implementasi [21].

Model RAD biasanya digunakan guna mempersingkat waktu yang dibutuhkan dan mendapat hasil dengan kualitas yang lebih baik dibandingkan dengan hasil yang diperoleh dari model lain. RAD sendiri merupakan gabungan dari berbagai teknik terstruktur dengan teknik *prototyping* dan teknik *joint application* untuk mempercepat pengembangan perangkat lunak [22]. Efisiensi dari segi waktu ini dibuktikan oleh berbagai penelitian yang telah dilakukan. Saraswati, dkk (2021) menyimpulkan keterlibatan pengguna melalui tahapan wawancara dan observasi yang mendetail, tahapan pengembangan sistem dapat diselesaikan dalam kurun waktu 3 bulan. Dari hal tersebut dapat dianalisis bahwa RAD cocok untuk pengembangan sistem payroll yang dikembangkan [3]. Hal tersebut disimpulkan karena skala dari perangkat lunak yang telah dikembangkan tidak terlalu besar, jadi cocok dengan karakteristik model RAD. Ari Setyati, Lega Anrima, Ainul R. I., dan Ifan dalam penelitian berjudul Pengembangan Aplikasi *Learning Course System* Kurteyki Berbasis Web dengan Metode RAD juga menunjukkan bahwa dengan menggunakan model *Rapid Application Development* pembuatan perangkat lunak menjadi lebih singkat dan cepat hanya dalam kurun waktu 90 hari [23].

Untuk membandingkan tingkat efisiensi model RAD dibandingkan dengan model pengembangan lain. Delima, dkk (2017) menemukan bahwa dalam mengukur efektifitas pengembangan menggunakan model RAD, ketika dibanding dengan model tradisional (*waterfall*) dapat dianalisa bahwa waktu pengembangan yang diperlukan ketika menggunakan RAD adalah 150 hari atau sekitar 30 minggu dari waktu yang disediakan yaitu 35 minggu [24]. Sementara ketika menggunakan model tradisional (*waterfall*) ditemukan bahwa waktu yang dihabiskan untuk pengembangan sistem sekitar 33 minggu [24]. Nur Nafara Rofiq dan Muniroh (2018) menyatakan bahwa pengembangan aplikasi dengan menerapkan model RAD memakan waktu yang cukup singkat. Suatu pengembangan perangkat lunak yang dimana umumnya memakan waktu minimal 180 hari, tetapi dengan model RAD, perangkat lunak tersebut dapat dirampungkan dalam waktu 60-90 hari [25]. Model RAD juga membantu tim pengembang untuk menghasilkan aplikasi yang sesuai dengan kebutuhan user. Widiyanto (2018) dalam penelitiannya menyatakan bahwa jika dibandingkan dengan 2 model lain yang digunakan, SIMPEG dengan metodologi RAD dibagi menjadi modul-modul yang lebih kecil dan sehingga dapat dikerjakan banyak tim secara bersama-sama dalam tengat waktu yang ditentukan [26]. Model ini melibatkan banyak tim, dan tiap-tiap tim diberikan tugas yang dengan level yang sama namun berbeda sesuai dengan pembagian modul perangkat lunak [26]. Dengan model RAD perangkat lunak yang dibangun menjadi lebih cepat tersampaikan ke pihak pengguna dikarenakan tidak diperlukannya waktu untuk menunggu fitur lain terselesaikan.

Selain untuk mengembangkan perangkat lunak berbasis *website* dan *mobile*, model RAD juga dapat digunakan untuk mengembangkan perangkat lunak berbasis IoT. Nalendra (2021) dalam penelitiannya menyebutkan bahwa model RAD dapat digunakan untuk mengembangkan sistem yang berbasis *Internet of Things* (IoT) [27]. Keberhasilan ini ditunjukkan dengan perangkat IoT yang diuji dengan pengukuran pompa irigasi terkontrol.

Aspek yang mempengaruhi Model RAD

Dalam penerapan model RAD pada pengembangan perangkat lunak terdapat pula berbagai aspek yang dapat mempengaruhi keberhasilan dan hambatan dalam

pengembangan perangkat lunak melalui penerapan model RAD [5] antara lain:

1. Aspek pendukung
 - Sistem tidak tertaut dengan sistem lain (*standalone*/bukan *embedded*)
 - Kinerja sistem tidak dipandang sebagai faktor penting.
 - Distribusi produk sempit.
 - Cangkupan terbatas.
 - Teknologi yang digunakan sudah dikenal oleh tim pengembang (tidak terlalu baru)
 - Sistem dapat dibagi menjadi beberapa modul dengan skala lebih kecil.
2. Aspek penghambat
 - Sistem tertaut kepada sistem yang lain (*embedded system*).
 - Kinerja menjadi faktor terpenting dalam sistem.
 - Distribusi produk luas.
 - Cangkupan sistem luas.
 - Sistem sulit atau untuk dibagi menjadi modul dengan skala yang lebih kecil.

Perbandingan Model RAD dengan Model Lain

Tabel 2. Perbandingan RAD dengan model lain

Model Rapid Application Development (RAD)	
Keuntungan	Kerugian
1) Produk yang dihasilkan akan lebih dapat diterima oleh konsumen [5],	1) Tingkat manajemen yang lebih kompleks [5],
2) Terdapat batasan-batasan pada sistem yang dibangun sehingga perubahan kebutuhan dapat ditampung [5],	2) Pengguna harus terlibat dalam seluruh siklus pengembangan [28],
3) Siklus yang dilalui oleh perangkat lunak lebih pendek jika didukung dengan penggunaan alat-alat RAD yang kuat [28],	3) Dibutuhkan personil tim pengembang yang terampil di tiap bidangnya [28],
4) Menghemat waktu, menghemat biaya, serta menghasilkan produk yang lebih berkualitas [5],	4) Sistem harus dapat dibagi menjadi modul-modul [5],
Model Tradisional (<i>Waterfall</i>)	
Keuntungan	Kerugian
1) Pengelolaan yang mudah, dikarenakan seluruh kebutuhan telah diidentifikasi dan di dokumentasi [29],	1) Tahapan yang linear, hingga tidak mungkin untuk kembali ke tahapan sebelumnya [29],
	2) Tidak fleksibel terhadap

- 2) Tahapan yang linier, sehingga mudah identifikasi dan mudah dipahami oleh seluruh tim dan *project owner* [29],
- 3) Cocok untuk proyek pengembangan skala kecil [30],
- 4) Kemajuan dalam proyek mudah ditandai [30],
- 3) Keteledoran atau ketidakmampuan menangkap Analisa kebutuhan terutama pada tahap *planning* dan *design* dapat bersifat fatal sehingga tidak dapat ditoleransi [29],
- 4) Dibutuhkan waktu yang cukup lama dari mulainya pengembangan sistem sampai dengan sistem dapat digunakan oleh *user* [5],
- 5) Sistem yang dikembangkan tidak memiliki kegunaan apabila belum semua tahapan belum diselesaikan seluruhnya [5],

Model Prototyping

Keuntungan	Kerugian
1) Mengikuti pendekatan <i>evolutionary</i> dan <i>iterative</i> [30],	1) Semakin lama waktu pengembangan, semakin kompleks pula sistem yang dikembangkan [28],
2) Dapat digunakan ketika kebutuhan <i>customer</i> masih belum diputuskan [30],	2) Personil tim dinilai akan memiliki beban yang lebih berat dari model lain [29],
3) Kesalahan dan pengulangan bisa diminimalisir karena adanya proses identifikasi yang terhadap <i>prototype</i> yang dibuat [28],	3) Terdapat biaya tambahan untuk pembuatan <i>prototype</i> awal [29],

4. PENUTUP

Kesimpulan

Berdasarkan dengan data dari jurnal serta penelitian yang telah ada yang kemudian dikumpulkan, dapat dikatakan bahwa model RAD memiliki keunggulan dalam hal efisiensi waktu untuk pengembangan perangkat lunak. Dibandingkan dengan model lain, model RAD dapat menghemat waktu selama 21 hingga 120 hari berdasarkan data yang telah dikumpulkan, tergantung dengan bentuk dan skala proyek pengembangan perangkat lunak. Model RAD juga mempermudah pengembangan sistem karena memiliki tahapan-tahapan yang sistematis dan cepat. Dibalik itu, model RAD juga memiliki aspek penunjang dan penghambat keberhasilan penerapan model dalam pengembangan perangkat lunak. Model RAD juga memiliki kelebihan yang lain disamping dari efisiensi waktu, tetapi model

RAD juga tak lepas dari kelemahan, walaupun kelemahan ini tidak terlalu signifikan jika dibandingkan dengan kelebihan dan hasil yang nantinya akan didapatkan. Jika dibandingkan dengan model *Waterfall* dan model *Prototyping*, model RAD memiliki kelebihan yang lebih signifikan dan kekurangan yang tidak terlalu berarti bila dibandingkan dengan kekurangan model *Waterfall* dan *Prototyping*.

Saran

Untuk pengembangan lebih lanjut, disarankan membandingkan model RAD dengan model-model yang lain dengan data-data yang lebih banyak lagi agar hasil penelitian lebih akurat.

5. REFERENSI

- [1] R. Parlita, D. Azizah, S. Latifah and B. D. Hadi, "Studi Literature Optimasi waktu dan Biaya pada Proyek Perangkat Lunak," : *Journal of Computer, Information System, & Technology Management*, vol. 4, no. 2, pp. 101-108, Oktober 2021.
- [2] R. Naz and M. N. A. Khan, "Rapid Application Development Techniques: A Critical Review," *International Journal of Software Engineering and Its Application*, vol. 9, no. 11, pp. 163-176, 2015.
- [3] N. W. S. Saraswati, N. W. Wardani, K. L. Maswari and I. D. M. K. Muku, "Rapid Application Development untuk Sistem Informasi Payroll Berbasis Web," *MATRIK : Jurnal Manajemen, Teknik Informatika Dan Rekayasa Komputer*, vol. 20, no. 2, pp. 213-224, 2021.
- [4] M. L. DESPA, "Comparative study on software development methodology," *Database Systems Journal*, vol. V, no. 3, pp. 37-56, 2014.
- [5] S. Aswati, M. S. Ramadhan, A. U. Firmansyah and K. Anwar, "STUDI ANALISIS MODEL RAPID APPLICATION DEVELOPMENT DALAM PENGEMBANGAN SISTEM INFORMASI," *MATRIK : Jurnal Manajemen, Teknik Informatika Dan Rekayasa Komputer*, vol. 16, no. 2, pp. 20-27, 2017.
- [6] O. Irnawati and G. B. A. Listianto, "Metode Rapid Application Development (RAD) pada Perancangan Website Inventory PT. SARANA ABADI MAKMUR BERSAMA (S.A.M.B)

- JAKARTA," *Jurnal Evolusi*, vol. 6, no. 2, pp. 12-18, 2018.
- [7] Tim Redaksi KBBI, "Kamus Besar Bahasa Indonesia. [Dalam Jaringan]," [Online]. Available: <https://kbbi.web.id/efisiensi>. [Accessed 6 April 2022].
- [8] R. Juliarto, "Dicoding," Dicoding, 28 April 2021. [Online]. Available: <https://www.dicoding.com/blog/apa-itu-efisien-pengertian-dan-contohnya/>. [Accessed 23 Maret 2022].
- [9] A. Mishra and D. Dubey, "A Comparative Study of Different Software Development Life Cycle Models in Different Scenarios," *International Journal of Advance Research in Computer Science and Management Studies (IJARCSMS)*, vol. 1, no. 3, pp. 64-69, 2013.
- [10] B. N. Hapshakh, D. N. Prasetyanti and P. D. Abda'u, "Implementasi Rapid Application Development dalam Pengembangan Aplikasi Pendaftaran Wisuda," *Journal of Innovation Information Technology and Application (JINITA)*, vol. 3, no. 1, pp. 72-81, June 2021.
- [11] R. Ferrari, "Writing narrative style literature reviews," *Medical Writing*, vol. 24, no. 4, 2015.
- [12] S. Kosasi, "Penerapan Rapid Application Development Dalam Sistem Perniagaan Elektronik Furniture," *Citec Journal*, vol. 2, no. 4, pp. 265-276, 2015.
- [13] A. Chusyairi, B. Rudiawan and E. Sasongko, "Layanan Laporan Kehilangan (E-Report) Dengan Metode Rapid Application Development," *Jurnal Telematika*, vol. 11, no. 1, pp. 52-54, Februari 2018.
- [14] A. D. Supriatna, "Designing library information system using rapid application development method," *IOP Conf. Ser.: Mater. Sci. Eng.*, 2018.
- [15] A. Andriani and E. Qurniati, "Sistem Informasi Penjualan Pada Toko Online Dengan Metode Rapid Application Development (RAD)," *Journal Spread (Sentra Penelitian Engineering dan Edukasi)*, vol. 10, no. 3, pp. 49-54, 2018.
- [16] A. Suryanto and M. I. Maliki, "Penerapan Model Rapid Application Development (RAD) Dalam Rancang Bangun Sistem Informasi Warga," *Infotek: Jurnal Informatika dan Teknologi*, vol. 5, no. 1, pp. 197-208, Januari 2022.
- [17] S. Al Ayubi and Y. Budiarti, "SISTEM INFORMASI E-KEPEGAWAIAN MENGGUNAKAN MODEL RAPID APPLICATION DEVELOPMENT (RAD) PADA YAYASAN BINA INSAN KAMIL JAKARTA," *JIKA (Jurnal Informatika)*, pp. 1-9, Februari 2022.
- [18] M. Y. Putra and R. W. R. Lolly, "Sistem Aplikasi Penjualan Souvenir Berbasis Web Menggunakan Metode Rapid Application Development (RAD)," *INFORMATION SYSTEM FOR EDUCATORS AND PROFESSIONALS*, vol. 5, no. 2, pp. 151-160, Juni 2021.
- [19] Y. D. Wijaya, "PENERAPAN METODE RAPID APPLICATION DEVELOPMENT (RAD) DALAM PENGEMBANGAN SISTEM INFORMASI DATA TOKO," *SITECH: JURNAL SISTEM INFORMASI DAN TEKNOLOGI*, vol. 3, no. 2, pp. 97-102, Desember 2020.
- [20] Suyanto and Andri, "IMPLEMENTASI RAPID APPLICATION DEVELOPMENT DALAM PENGEMBANGAN APLIKASI PELAPORAN KERUSAKAN JALAN," *JUPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 05, no. 02, pp. 89-97, Desember 2020.
- [21] A. Baihaqi and Tumini, "Penerapan Metode Rapid Application Development (RAD) Dalam Pengembangan Sistem Pemesanan Menu Berbasis Android," *Jurnal informasi dan Komputer*, vol. 9, no. 2, pp. 95-102, 2021.
- [22] J. Parhusip, A. C. Saputra, C. Handika and I. P. P., "PENGEMBANGAN APLIKASI BIMBINGAN KONSELING PADA SMK NEGERI 1 MUARA TEWEH MENGGUNAKAN METODE RAPID APPLICATION DEVELOPMENT (RAD)," *Jurnal Keilmuan dan Aplikasi Bidang Teknik Informatika*, vol. 16, no. 1, pp. 61-72, Januari 2022.
- [23] A. Setyati, L. A. Setiawati, A. R. Ichwanto and I. Prihandi, "Pengembangan Aplikasi Learning Course System Kurteyki Berbasis Web Dengan Metode RAD," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 8, no. 3, pp. 1483-1495, September 2021.

- [24] R. Delima, H. B. Santosa and J. Purwadi, "Development of Dutatani Website Using Rapid Application Development," *IJITEE (INTERNATIONAL JOURNAL OF INFORMATION TECHNOLOGY AND ELECTRICAL ENGINEERING)*, vol. 1, no. 2, pp. 36-44, June 2017.
- [25] N. N. Rofiq and S. Muniroh, "IMPLEMENTASI RAPID APPLICATION DEVELOPMENT MODEL PADA APLIKASI PENDAFTARAN RAWAT JALAN BERBASIS WEB," *Prosiding Seminar Nasional Informatika dan Sistem Informasi*, vol. 3, no. 1, pp. 28-38, Mar-Jun 2018.
- [26] W. W. Widiyanto, "ANALISA METODOLOGI PENGEMBANGAN SISTEM DENGAN PERBANDINGAN MODEL PERANGKAT LUNAK SISTEM INFORMASI KEPEGAWAIAN MENGGUNAKAN WATERFALL DEVELOPMENT MODEL, MODEL PROTOTYPE, DAN MODEL RAPID APPLICATION DEVELOPMENT (RAD)," *Jurnal INFORMA Politeknik Indonusa Surakarta*, vol. 4, no. 1, pp. 34-40, 2018.
- [27] A. K. Nalendra, "Rapid Application Development (RAD) model method for creating an agricultural irrigation system based on internet of things," *IOP Conf. Ser.: Mater. Sci. Eng.*, 2021.
- [28] D. S. Budi, T. A. Y. Siswa and H. Abijono, "Analisis Pemilihan Penerapan Proyek Metodologi Pengembangan Rekayasa Perangkat Lunak," *TEKNIKA*, vol. 5, no. 1, pp. 24-31, November 2016.
- [29] M. Bolung and H. R. K. Tampangela, "ANALISA PENGGUNA METODOLOGI PENGEMBANGAN PERANGKAT LUNAK," *Jurnal ELTIKOM*, vol. 1, no. 1, pp. 1-10, Juni 2017.
- [30] S. M. Salve, S. N. Samreen and N. Khatri-Valmik, "A Comparative Study on Software Development Life Cycle Models," *International Research Journal of Engineering and Technology (IRJET)*, vol. 05, no. 02, pp. 696-700, February 2018.