

Review Analisis Platform CICD dengan Metode Systematic Literature Review

Rizqi Agung Permana¹, Sucitra Sahara², Marlina³

¹STMIK Antar Bangsa, ^{2,3}Universitas Bina Sarana Informatika

¹e-mail: rizqiagung@gmail.com, ²e-mail: sucitra.scr@bsi.ac.id, ³marlina.mln@bsi.ac.id

Diterima	Direvisi	Disetujui
19-09-2022	09-11-2022	10-01-2023

Abstrak - Banyak aplikasi yang dibentuk, mulai dari aplikasi mobile maupun berupa website. Selain itu dengan adanya pandemi juga turut mendorong pertumbuhan iklim digital di segala sektor. Hampir setiap organisasi baik industri, pendidikan maupun pemerintahan terus berlomba untuk berinovasi menciptakan dan menggunakan teknologi digital. Fitur dan layanan aplikasi pun terus berkembang dan beradaptasi dengan experience pengguna. Perubahan atau release terhadap aplikasi pun di tuntut semakin cepat dan zero error. Proses deployment ke production server tersebut menjadi semakin sering dilakukan. Sehingga kebutuhan akan tools atau platform CICD sangat tinggi. Berdasarkan hasil penelitian Systematic Literature Review terhadap jurnal scopus yang terbit antara tahun 2019 sampai 2021 yang memiliki peringkat di scimagojr Q1 sampai Q4 dengan keyword CICD atau DevOps. Terdapat 6 paper yang menyebutkan platform Jenkins. Sedangkat jika keyword di rubah menjadi Jenkins dengan tipe journal terdapat 193 paper yang mencantumkan Jenkins. Kemudian untuk metode deployment rollout di pakai dalam 10 paper penelitian dan menjadi metode terbanyak yang di pakai. Sedangkan untuk arsitektur applikasi yang paling banyak di pakai adalah microservice dengan 20 paper.

Kata Kunci: Devops, Platform CICD, Systematic Literature Review

Abstract - Many applications are formed, ranging from mobile applications to websites. In addition, the pandemic has also contributed to the growth of the digital climate in all sectors. Almost every organization in industry, education and government continues to compete to innovate, create and use digital technology. Application features and services continue to develop and adapt to the user experience. Changes or releases to applications are also demanded to be faster and have zero errors. The deployment process to the production server is becoming more and more frequent. So the need for CICD tools or platforms is very high. Based on the results of a review analysis research on Scopus journals published between 2019 and 2021 that rank in scimagojr Q1 to Q4 with the keywords CICD and/or DevOps. There are 6 papers that mention the Jenkins platform. Meanwhile, if the keyword is changed to Jenkins with journal type, there are 193 papers that include Jenkins. Then the deployment rollout method was used in 10 research papers and became the most used method. As for the application architecture, the most widely used is microservice with 20 papers.

Keywords: Review Analysis, Platform, Systematic Literature

PENDAHULUAN

Semakin hari banyak aplikasi yang lahir baik aplikasi mobile maupun website. Selain itu adanya pandemi juga turut mendorong pertumbuhan iklim digital di segala sektor. Hampir setiap organisasi baik industri, pendidikan maupun pemerintahan terus berlomba untuk berinovasi menciptakan dan menggunakan teknologi digital. Fitur dan layanan aplikasi pun terus berkembang. Perubahan maupun penambahan fitur sebuah aplikasi sudah menjadi aktivitas sehari-hari. Hal ini dapat berarti semakin

banyak aplikasi atau software yang perlu maintenance oleh tim operator terutama dalam hal pengembangan atau release fitur baru. Dimana setiap aplikasi dituntut untuk selalu agile dalam menghadapi perubahan maupun permintaan pengguna. Aplikasi yang cenderung sangat beragam dan terdiri dari berbagai alat, teknologi komersial atau sumber terbuka, oleh karena itu mengelola siklus hidup pengembangan aplikasi menjadi sulit karena kompleksitas tersebut. Menghadapi tugas-tugas yang menakutkan dan kompleks tersebut, salah satu solusinya adalah menggunakan pendekatan



DevOps(Ferdian et al. 2021). Siklus hidup pengembangan sistem (SDLC) merupakan metodologi untuk merancang, membangun, mengembangkan dan memelihara informasi dan proses sistem(Tohirin et al. 2020). Dalam proses SDLC aplikasi akan di design lalu dilakukan build kemudian testing dan di deploy. Selanjut tahapan SDLC ini akan terus terulang dan akan terdapat dua kubu di dalam tim aplikasi yaitu subtim developer dan subtim operation (Teixeira et al. 2020). Dua bagian subtim tersebut perlu melakukan integrasi dan kolaborasi agar setiap terjadi release atau peningkatan fitur bisa berjalan dengan baik dan tidak terjadi kesalahan komunikasi. Karena ketika dua tim terpisah melakukan Dev dan Ops, masalah dapat muncul dan kurangnya sinergi dapat terjadi sehingga dapat mempengaruhi atau membahayakan kualitas aplikasi.

CICD atau kepanjangan dari Continuous Integration Continuous Delivery merupakan salah satu pendekatan DevOps yang sangat tren beberapa tahun ini (Ferdian et al. 2021). DevOps pun memiliki peran yang sangat penting dalam sebuah aplikasi(Hemon-Hildgen, Rowe, and Monnier-Senicourt 2020). Untuk membantu proses CICD tim devops banyak sekali di hadapkan dengan berbagai macam platform khususnya di sisi Continuous Integration. Contohnya Jenkins, Gitlab CI, Circle CI dan lain-lain. Pemilihan platform yang tepat menjadi sangat penting karena dapat mengurangi learning curve serta meningkatkan agility sebuah aplikasi. Dari platform CICD yang digunakan kemudian akan ditentukan terkait penggunaan metode deployment. Apakah menggunakan basic deployment, Canary, rolling, A/B testing dan lain-lain. Selanjutnya akan di ketahui arsitektur server yang sesuai dengan metode yaitu monolith atau microservice.

Data-data yang di kumpulkan dalam penelitian ini di ambil dari jurnal scopus dengan waktu terbit dari tahun 2019 sampai 2020. Data jurnal tersebut akan di identifikasi dengan metode Systematic Literature Review (SLR). Dengan demikian di proses review CICD dapat bersifat objektive dan jauh dari sifat subjektif sehingga hasil dari penelitian ini di harapkan dapat memberikan rekomendasi platform CICD kepada tim DevOps, metode deployment dan rancangan arsitektur.

METODOLOGI PENELITIAN

Menjelaskan kronologis penelitian, termasuk desain penelitian, prosedur penelitian (dalam bentuk algoritma, Pseudocode atau lainnya), bagaimana untuk menguji dan akuisisi data. Deskripsi dari program penelitian harus didukung referensi, sehingga penjelasan tersebut dapat diterima secara ilmiah.

Metode penelitian yang digunakan adalah Systematic Literature Review (SLR). Dalam SLR data penelitian akan di kumpulkan dan kemudian di buat evaluasi terkait topik atau subjek tertentu

(Triandini et al. 2019). SLR dapat mengurangi duplikasi tema penelitian dan mengoptimalkan hasil dari penelitian-penelitian sebelumnya(Lame 2019). SLR dapat meminimalkan bias dalam tinjauan sistematis untuk identifikasi, seleksi, sintesis dan ringkasan singkat dari studi yang berbeda(Khan and Qureshi 2020).

Tujuan dari tinjauan SLR adalah untuk mengidentifikasi semua bukti empiris yang sesuai dengan kriteria inklusi yang telah ditentukan sebelumnya untuk menjawab pertanyaan penelitian atau hipotesis tertentu(Snyder 2019). Sebagian besar tinjauan literatur berfungsi sebagai bagian dari artikel penelitian utama yang memberikan landasan teoritis untuk studi utama yang menjadi subjek artikel(Okoli and Schabram 2010).

a. Objek penelitian

Dalam penelitian ini objek yang akan di teliti adalah penelitian Continuous Integration Continuous Delivery (CICD).

Mengingat perkembangan terhadap kebutuhan aplikasi yang semakin meningkat dan aplikasi semakin di tuntutan untuk lebih flexible terhadap perubahan. Kebutuhan akan penerapan CICD oleh DevOps pun menjadi sangat penting(Ferdian et al. 2021). Banyaknya platform CICD yang tersedia, berbagai metode deployment dan arsitektur sebuah aplikasi menjadi point penting dalam mengambil keputusan.

b. Research Question

Research Question merupakan pertanyaan-pertanyaan terkait dengan topik pembahasan atau object penelitian yang di buat berdasarkan kebutuhan penelitian. Berikut pertanyaan-pertanyaan yang peneliti buat terkait objek CICD:

RQ1. Platform Continuous Integration Continuous Delivery (CICD) apa yang paling banyak di pakai dalam DevOps di tahun 2019–2020?

RQ2. Metode deployment apa yang digunakan dalam Continuous Integration Continuous Delivery (CICD)?

RQ3. Arsitektur aplikasi seperti apa yang paling banyak menggunakan Continuous Integration Continuous Delivery (CICD)?

c. Search Process

Proses pencarian data di lakukan untuk mengumpulkan sumber-sumber yang relevan terkait dengan Research Question(Triandini et al. 2019). Dalam hal ini peneliti menggunakan software publich or perish untuk melakukan pencarian dari jurnal scopus. Publich or perish merupakan program perangkat lunak yang mengambil dan menganalisis kutipan akademik.

d. Inclusion and Exclusion Criteria

Proses Inclusion dan exclusion menjadi salah satu tahapan yang cukup penting. Dalam tahap ini data yang terkumpul dari prose pencarian akan di lakukan filtering sesuai dengan kriteria yang terkait dengan objek penelitian. Berikut beberapa kriteria harus

terdapat dalam jurnal yang akan menjadi bahan penelitian:

1. Jurnal yang digunakan adalah yang terbit dengan rentang waktu 2019 sampai 2020
2. Jurnal yang digunakan hanya jurnal yang di ambil dari scopus.
3. Jurnal yang digunakan hanya jurnal yang berhubungan dengan Continuous Integration Continuous Delivery (CICD).

e. Quality Assesment

Tahap quality assessment dilakukan untuk mengevaluasi data atau jurnal yang sesuai dengan pertanyaan berikut:

QA1. Apakah pada paper jurnal menuliskan platform yang digunakan dalam proses Continuous Integration Continuous Delivery (CICD)??

QA2. Apakah pada paper jurnal terdapat point methode deployment apa yang digunakan dalam penerapan CICD ?

QA3. Apakah pada paper jurnal menuliskan aritektur aplikasi yang di deploy dengan CICD?

Hasil dari jawaban pertanyaan terkait kreteria diatas akan ada dua kemungkinan yaitu ya atau tidak. Sehingga dalam pemrosesan jurnal yang sesuai dengan Quality Assesment, setiap jurnal akan di beri label Y jika sesuai dan label T jika tidak sesuai.

f. Data Collection

Dalam tahap data collection ini data yang di peroleh akan di kumpulkan dan kemudian akan di lakukan analisa(Apriliani et al. 2020). Data yang dikumpulkan dalam penilitan ini adalah data primer.

g. Analysis dan Report

Setelah melakukan data collection penting untuk mempertimbangkan bagaimana artikel akan digunakan dalam melakukan analisis yang tepat(Snyder 2019). Data yang terkumpul akan di analisa secara mendalam sesuai mendalam berdasarkan konten data atau jurnal. Kemudian akan di buatkan maaping sebaran serta grafik secara visual untuk memperesentasikan kondisi penerapan Continuous Integration Continuous Delivery (CICD) pada Devops.

HASIL DAN PEMBAHASAN

1. Search procces

Dalam melakukan proses pencarian jurnal penulis menggunakan perangkat lunak publish or perish dengan keyword cicd, DevOps, Jenkins dan publication name journal seperti pada gambar berikut:



Gambar 1. Search proses dengan Publis or Perish

Selanjutnya untuk melengkapai atribut dalam jurnal, hasil pencarian dari publis or perish akan di export dalam format reference manager (.ris). File dengan extensi ris tersebut akan di import ke dalam perangkat lunak mendeley guna dilakukan update attribute.

- a. Hasil Seleksi Inclusion and Exclusion Criteria Berdasar hasil pencarian melalui Publis or Perish, paper jurnal yang di dapat 63 jurnal. Dari kumpulan jurnal yang di dapet tersebut di filter Kembali sesuai dengan exclisiton dan exclition yang telah di definisikan sebelumnya. Selain itu peneliti juga menambahkan filtering terkait peringkat jurnal yang merujuk pada Scimagojr. Dimana Scimagojr merupakan salah satu situs yang melakukan rangking , sitasi terhadap jurnal terindeks Scopus.
- b. Hasil Quality Assesment

Tabel 1. Hasil quality Assesment Jurnal

No	Title	Tahun	Ranking	Q1	Q2	Q3
1	Model-based deployment of secure multi-cloud applications	2019	Q3	T	Y	Y
2	A systematic literature review on DevOps capabilities and areas	2020	Q3	T	Y	Y
3	Orchestrating automation and sharing in DevOps teams: a revelatory case of job satisfaction factors, risk and work conditions	2020	Q1	T	Y	Y
4	Network Management 2030: Operations and Control of Network 2030 Services	2020	Q2	T	T	T
5	A hybrid DevOps process supporting software reuse: A pilot project	2020	Q3	T	Y	Y
6	DevOpRET: Continuous reliability testing in DevOps	2020	Q3	T	T	T
7	Design on deployment of microservices on container-based cloud platform	2020	Q4	T	Y	Y
8	Profile and treatment of chronic coronary syndromes in European Society of Cardiology member countries: The ESC EORP CICD-LT registry	2020	Q1	T	T	T
9	A new autonomous inflow control device designed for a loose sand oil reservoir with bottom water	2019	Q1	T	T	T

10	Adopting DevOps in the real world: A theory, a model, and a case study	2019	Q1	Y	Y	Y
11	Sinplafut: A microservices - Based application for soccer training	2019	Q4	Y	Y	Y
12	An attempt to adopt DevOps on embedded system development: Empirical evidence	2019	Q4	Y	Y	Y
13	Multicriteria decision-making taxonomy for DevOps challenging factors using analytical hierarchy process	2020	Q3	T	T	T
14	DevOps for IoT Systems: Fast and Continuous Monitoring Feedback of System Availability	2020	Q1	T	Y	Y
15	Adopting integrated application lifecycle management within a large-scale software company: An action research approach	2019	Q1	T	T	Y
16	A novel Security-by-Design methodology: Modeling and assessing security by SLAs with a quantitative approach	2020	Q1	T	Y	Y
17	Understanding how DevOps aligns development and operations: a tripartite model of intra-IT alignment	2020	Q1	T	Y	Y
18	A Systematic Mapping Study on Microservices Architecture in DevOps	2020	Q1	T	Y	Y
19	Two experiments for evaluating the impact of Hamcrest and AssertJ on assertion development	2020	Q1	T	T	T
20	CORRIGENDUM: Profile and treatment of chronic coronary syndromes in European Society of Cardiology member countries: The ESC EORP CICD-LT registry (European Journal of Preventive Cardiology , (2020), 27, 8, 10.1177/2047487320912491)	2020	Q1	T	T	T
21	An automated framework for continuous development and testing of access control systems	2020	Q3	Y	Y	Y
22	ICSSP 2018—Special issue introduction	2019	Q3	T	T	T
23	Extending reference architecture of big data systems towards machine learning in edge computing environments	2020	Q1	T	T	Y
24	Improving microservice-based applications with runtime placement adaptation	2019	Q2	T	Y	Y
25	Enabling continuous integration in a formal methods setting	2020	Q2	Y	Y	Y
26	Architecting Microservices: Practical Opportunities and Challenges	2020	Q1	Y	Y	Y
27	Continuous Delivery of Customized SaaS Edge Applications in Highly Distributed IoT Systems	2020	Q1	T	Y	Y
28	Good practices for the adoption of DataOps in the software industry	2020	Q4	T	T	T
29	Software Quality Assurance in INDIGO-DataCloud Project: a Converging Evolution of Software Engineering Practices to Support European Research e-Infrastructures	2020	Q1	Y	Y	Y

- c. Hasil Analisa dan Reporting
Berdasarkan data hasil quality assesment selanjutnya di lakukan analisa untuk di ambil kesimpulan sesuai dengan point Research Question (RQ).



Gambar 2. Proses Analisa dan Report

Pendekatan devops melalui Continuous Integration Continuous Delivery (CICD) sudah banyak di adopsi dalam hal mempermudah pengelolaan aplikasi oleh developer maupun tim operation. Walaupun ada beberapa issue yang muncul terumata dari segi keamanan. Namun Continuous Integration Continuous Delivery (CICD) telah terbukti banyak membantu baik dari segi waktu pengerjaan maupun penghematan biaya operation.

RQ1. Platform Continuous Integration Continuous Delivery (CICD) apa yang paling banyak di pakai dalam DevOps di tahun 2019–2020?

Berdasarkan hasil SLR yang bersumber dari jurnal Scopus dengan keyword CICD maupun DevOps.

Platform jenkins menjadi salah satu yang paling banyak muncul. Jenkins merupakan salah satu tool CI yang dikembangkan oleh Kohsuke Kawaguchi dengan nama awal Hudson pada akhir 2004 ketika masih bekerja di Sun(Priera and Ganefi 2017). Kemudian pada tahun 2011 tim pengembang memutuskan untuj mengubah nama Hudson menjadi Jenkins. Jenkins memiliki banyak sekali plugin yang bisa di install sehingga sangat mudah untuk mengintegrasikan dengan tools lainnya. Selain itu Jenkins tidak hanya support ke VCS tertentu. Layanan parameter pipeline yang

sangat dinamis juga sangat membantu ketika melakukan build atau compile dengan parameter.

RQ2. Metode deployment apa yang digunakan dalam Continuous Integration Continuous Delivery (CICD)?

Metode deployment yang paling banyak di pakai dalam jurnal yang direview adalah rollout deployment strategi. Rollout deployment merupakan salah satu metode deployment yang di lakukan secara bertahap(Xia 2019). Tujuannya adalah agar jika terjadi error tidak semua server mengalami error dan dapatsegera di lakukan rollback. Hal ini bisa mengurangi resiko experien pengguna yang buruk.

Dalam prakteknya sebuah aplikasi yang di deploy dengan metode rollout akan memiliki dua versi yang berjalan di server yang berbeda. Sehingga pengaturan versi sourcode contohnya dalam hal tag maupun branch sangat membantu. Sourcode control management yang baik menjadi salah satu kunci dalam implementasi metode tersebut.

RQ3. Arsitektur aplikasi seperti apa yang paling banyak menggunakan Continuous Integration Continuous Delivery (CI/CD)?

Dalam jurnal yang di review, arsitektur aplikasi yang digunakan paling banyak adalah *microservice*. Baik aplikasi berbasis web maupun mobile menggunakan arsitektur *microservice*. Walaupun begitu masih ada beberapa jurnal yang membandingkan implementasi *microservice* dengan *monolith*. Beberapa jurnal juga membahas *methodologi* untuk migrasi dari *monolith* ke *microservices*. Arsitektur *microservice* sendiri merupakan alternatif arsitektur yang lebih terukur dan lebih fleksible (Munawar and Hodijah 2018). Dalam arsitektur *microservice* sebuah aplikasi akan dikembangkan dalam *service-service* kecil yang terpisah secara *service* namun saling berkomunikasi antara satu dan lainnya (Mufriзал and Indarti 2019). Berikut kelebihan arsitektur *monolith*:

1. Setiap *service* aplikasi akan bersifat *independent*.
2. Maintenance aplikasi menjadi lebih mudah.
3. Setiap *service* bisa menggunakan Bahasa pemrograman yang berbeda.
4. Kemudahan dalam scalability.

KESIMPULAN

Berdasarkan hasil penelitian analisis review terhadap jurnal scopus yang terbit antara tahun 2019 sampai 2021 yang memiliki peringkat di scimagojr Q1 sampai Q4 dengan keyword CI/CD dan /atau DevOps. Terdapat 6 paper yang menyebutkan platform Jenkins. Sedangkat jika keyword di rubah menjadi Jenkins dengan type journal terdapat 193 paper yang mencantumkan Jenkins. Kemudian untuk metode deployment rollout di pakai dalam 10 paper penelitian dan menjadi metode terbanyak yang di pakai. Sedangkan untuk arsitektur aplikasi yang paling banyak di pakai adalah *microservice* dengan 20 paper. Menurut hasil dari penelitian dapat disimpulkan bahwa platform Jenkins menjadi platform paling flexible dan banyak diadopsi. Untuk penelitian selanjutnya perlu di lakukan perluasan jangkauan tahun serta keyword pencarian dan listing update platform terbaru Selain itu penelitian ini hanya di fokuskan pada jurnal scopus.

REFERENSI

- Apriliani, Alma, Muhammad Budhiluhoer, Asep Jamaludin, and Kamal Prihandani. 2020. "Systematic Literature Review Kepuasan Pelanggan Terhadap Jasa Transportasi Online." *Systematics* 2(1): 12.
- Ferdian, Sedy et al. 2021. "Continuous Integration and Continuous Delivery Platform Development of Software Engineering and Software Project Management in Higher Education." *Jurnal Teknik Informatika dan Sistem Informasi* 7(1): 59–70.
- Hemon-Hildgen, Aymeric, Frantz Rowe, and Laetitia Monnier-Senicourt. 2020. "Orchestrating Automation and Sharing in DevOps Teams: A Revelatory Case of Job Satisfaction Factors, Risk and Work Conditions." *European Journal of Information Systems*: 474–99. https://api.elsevier.com/content/abstract/scopus_id/85088032300.
- Khan, Nohman, and Muhammad Imran Qureshi. 2020. "A Systematic Literature Review on Online Medical Services in Malaysia." *International journal of online and biomedical engineering* 16(6): 107–18.
- Lame, Guillaume. 2019. "Systematic Literature Reviews: An Introduction." *Proceedings of the International Conference on Engineering Design, ICED 2019-Augus(AUGUST)*: 1633–42.
- Marcoulides, George a. 2005. 100 Journal of the American Statistical Association *Discovering Knowledge in Data: An Introduction to Data Mining: Discovering Knowledge in Data: An Introduction to Data Mining*.
- Mufriзал, Rizki, and Dina Indarti. 2019. "Refactoring Arsitektur Microservice Pada Aplikasi Absensi PT. Graha Usaha Teknik." *Jurnal Nasional Teknologi dan Sistem Informasi* 5(1): 57–68.
- Munawar, Ghifari, and Ade Hodijah. 2018. "Analisis Model Arsitektur Microservice Pada Sistem Informasi DPLK." *Sinkron: Jurnal dan Penelitian Teknik Informatika* 3(1): 232–39.
- Okoli, Chitu, and Kira Schabram. 2010. "Working Papers on Information Systems A Guide to Conducting a Systematic Literature Review of Information Systems Research." *Working Papers on Information Systems* 10(2010).
- Priera, Jaga My, and Robi Tanzil Ganefi. 2017. "Automatic Deployment System Dengan Menggunakan Metode Continuous Integration Di Kakatu." *Jurnal Ilmiah Komputer dan Informatika*.
- Snyder, Hannah. 2019. "Literature Review as a Research Methodology: An Overview and Guidelines." *Journal of Business Research* 104(August): 333–39.

- <https://doi.org/10.1016/j.jbusres.2019.07.039>.
- Teixeira, Daniel et al. 2020. "A Systematic Literature Review on DevOps Capabilities and Areas." *International Journal of Human Capital and Information Technology Professionals* 11(3): 1–22. https://api.elsevier.com/content/abstract/scopus_id/85083568991.
- Tohirin, Tohirin, Sri Farida Utami, Septian Rheno Widiyanto, and Widhy Al Mauludyansah. 2020. "Implementasi DevOps Pada Pengembangan Aplikasi E-Skrining Covid-19." *Multinetics* 6(1): 15–20.
- Triandini, Evi et al. 2019. "Metode Systematic Literature Review Untuk Identifikasi Platform Dan Metode Pengembangan Sistem Informasi Di Indonesia." *Indonesian Journal of Information Systems* 1(2): 63.
- Xia, T. 2019. "Safe Velocity: A Practical Guide to Software Deployment at Scale Using Controlled Rollout." *Proceedings - 2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice, ICSE-SEIP 2019*: 11–20. https://api.elsevier.com/content/abstract/scopus_id/85072115479.