

Implementasi CI/CD Pada Proyek E-RBA

1st Argian Raditya Putra

Fakultas Ilmu Terapan
Universitas Telkom

Bandung, Indonesia

argianradityaputra21@gmail.com

2nd Hariandi Maulid

Fakultas Ilmu Terapan
Universitas Telkom

Bandung, Indonesia

hmaulid@telkomuniversity.ac.id

3rd Muhammad Purwadi

Chief Operation Officer
PT Tabel Data Informatika

Bandung, Indonesia

purwadi@tabeldata.com

Abstrak—Penelitian ini menyajikan sebuah studi tentang implementasi Continuous Integration/Continuous Deployment (CI/CD) pipeline di lingkungan kantor untuk mendukung pengembangan proyek E-RBA dengan *source code* yang kompleks. CI/CD merupakan metodologi pengembangan perangkat lunak yang mengintegrasikan integrasi kode, dan otomatisasi proses deploy guna meningkatkan efisiensi dan kualitas pengembangan.

Penelitian ini fokus pada proyek E-RBA yang merupakan proyek aplikasi web skala menengah yang sedang dikembangkan oleh tim di kantor PT Tabel Data Informatika. Tujuan dari penelitian ini adalah untuk mengurangi waktu siklus pengembangan, meningkatkan kualitas perangkat lunak, serta mencapai proses deploy yang lebih stabil dan konsisten melalui implementasi CI/CD pipeline.

Kata Kunci — PT Tabel Data Informatika, *DevOps*, *CI/CD*, *Gitlab*, *Kubernetes*.

I. PENDAHULUAN

A. Latar Belakang

Dalam era teknologi informasi yang terus berkembang pesat, perusahaan konsultan seperti PT Tabel Data Informatika memainkan peran penting dalam menyediakan solusi teknologi informasi yang inovatif dan efektif bagi klien-klien mereka. Pengembangan layanan digital dan komunikasi, khususnya pembuatan aplikasi berbasis web dan mobile, telah menjadi aspek kritis bagi perusahaan untuk memenuhi kebutuhan pelanggan dan tetap bersaing di pasar yang semakin kompetitif.

Salah satu proyek unggulan PT Tabel Data Informatika adalah proyek E-RBA, sebuah aplikasi yang bertujuan untuk mengelola Rencana Anggaran Belanja sebuah organisasi. Namun, saat ini, proses pengembangan aplikasi E-RBA masih mengandalkan alur manual dalam berbagai tahapan, termasuk penulisan kode, build aplikasi, deployment ke dalam server, dan monitoring aplikasi yang sudah berjalan. Ketergantungan pada proses manual ini menimbulkan beberapa masalah, seperti kesulitan dalam menghindari duplikasi tugas, penundaan rilis produk, dan ketidakmampuan untuk bersaing secara efisien dengan pesaing di pasar yang terus bergerak dinamis.

Untuk meningkatkan efisiensi dan kualitas pengembangan aplikasi E-RBA, penelitian ini menawarkan solusi berupa penggunaan metodologi Continuous Integration/Continuous Deployment (CI/CD). CI/CD adalah pendekatan pengembangan perangkat lunak yang mengintegrasikan pengujian otomatis, integrasi kode, dan otomatisasi proses deploy guna meningkatkan efisiensi, keandalan, dan kualitas pengembangan aplikasi. Dengan

demikian, melalui penerapan CI/CD menggunakan *Gitlab CI*, diharapkan tim terkait proyek E-RBA dapat lebih cepat dan konsisten dalam men-deliver aplikasi sesuai dengan standar perusahaan PT Tabel Data Informatika, serta dapat memanfaatkan hasil penelitian ini untuk proyek-proyek masa depan di perusahaan tersebut.

Penerapan CI/CD dalam proyek E-RBA di PT Tabel Data Informatika akan memberikan manfaat yang signifikan bagi perusahaan dan klien-klien mereka. Dengan otomatisasi proses deploy menggunakan *Gitlab CI*, tim pengembang dapat mengurangi kesalahan manusia dan mempercepat siklus pengembangan aplikasi. Pengujian otomatis yang terintegrasi dalam CI/CD pipeline akan membantu meningkatkan kualitas perangkat lunak dengan mendeteksi masalah lebih awal dan memungkinkan perbaikan cepat sebelum perilis. Hal ini akan mengurangi risiko kesalahan yang mungkin terjadi setelah aplikasi sudah berada di produksi.

Selain itu, CI/CD juga akan menciptakan lingkungan pengembangan yang lebih kolaboratif dan transparan. Tim pengembang akan dapat dengan mudah berbagi kode dan berkolaborasi dalam mengatasi masalah yang muncul. Proses integrasi kode yang terotomatisasi akan memastikan bahwa setiap kontribusi kode telah diuji secara otomatis dan sesuai dengan standar perusahaan, mengurangi potensi konflik dan meningkatkan kualitas kode secara keseluruhan.

Dengan adopsi CI/CD, PT Tabel Data Informatika akan dapat lebih responsif terhadap perubahan kebutuhan klien dan pasar. Kemampuan untuk secara cepat dan konsisten menerapkan perubahan dan memperbarui aplikasi akan memungkinkan perusahaan untuk menghadapi persaingan dengan lebih baik dan memberikan solusi yang lebih inovatif kepada klien. Klien juga akan merasa lebih percaya dengan kemampuan perusahaan untuk memberikan layanan yang lebih handal dan terus diperbarui.

Dalam jangka panjang, implementasi CI/CD di proyek E-RBA akan berdampak positif pada seluruh organisasi PT Tabel Data Informatika. Budaya continuous improvement dan inovasi akan semakin tertanam dalam setiap tim pengembang. Keahlian dalam menggunakan CI/CD akan menjadi aset berharga bagi perusahaan dan dapat diaplikasikan dalam proyek-proyek lain, meningkatkan efisiensi dan kualitas pengembangan di seluruh perusahaan.

B. Kontribusi Penelitian

Kontribusi penelitian ini diharapkan akan memberikan dampak yang signifikan bagi PT Tabel Data Informatika dan tim-tim yang terlibat dalam proyek E-RBA. Dengan

menghasilkan program otomatisasi menggunakan Gitlab CI yang sesuai dengan standar perusahaan, penelitian ini akan membawa efisiensi dan efektivitas baru dalam pengembangan aplikasi. Program otomatisasi ini akan mengurangi waktu dan usaha yang dibutuhkan untuk melakukan deployment, sehingga tim pengembang dapat lebih fokus pada aspek-aspek kreatif dan inovatif dalam pembuatan aplikasi. Selain itu, program otomatisasi ini akan memastikan konsistensi dalam proses deploy, mengurangi risiko kesalahan manusia, dan meningkatkan kualitas dan keandalan aplikasi yang dihasilkan.

Selain memberikan manfaat langsung bagi proyek E-RBA, hasil dari penelitian ini juga akan memiliki dampak jangka panjang bagi PT Tabel Data Informatika secara keseluruhan. Program otomatisasi yang dihasilkan dapat digunakan kembali dalam proyek-proyek berikutnya di perusahaan, sehingga perusahaan dapat lebih efisien dalam pengembangan aplikasi dan menghemat waktu serta sumber daya. Dengan adanya program otomatisasi yang telah teruji dan terbukti berhasil, PT Tabel Data Informatika juga akan memiliki keunggulan kompetitif dalam menawarkan solusi teknologi informasi kepada klien.

Selain itu, penelitian ini akan membantu membentuk budaya continuous improvement dan inovasi di PT Tabel Data Informatika. Penggunaan CI/CD dalam pengembangan proyek E-RBA akan menjadi contoh bagaimana pendekatan ini dapat meningkatkan kualitas dan kecepatan pengembangan. Tim pengembang akan terbiasa dengan metodologi ini dan akan terus menerapkannya dalam proyek-proyek masa depan. Dengan demikian, penelitian ini akan mendorong perusahaan untuk terus berinovasi dan beradaptasi dengan perubahan teknologi informasi yang terus berkembang.

Sebagai kontribusi ilmiah, penelitian ini akan menyumbangkan pengetahuan dan wawasan baru dalam pengembangan perangkat lunak berbasis CI/CD. Hasil penelitian ini dapat menjadi referensi dan panduan bagi perusahaan-perusahaan lain yang ingin mengadopsi CI/CD dalam proses pengembangan aplikasi mereka. Selain itu, penelitian ini juga akan menjadi kontribusi positif dalam perkembangan ilmu pengetahuan dan teknologi di bidang pengembangan perangkat lunak.

Dengan demikian, penelitian ini tidak hanya berfokus pada keberhasilan proyek E-RBA, tetapi juga memiliki potensi untuk mengubah paradigma dan praktik pengembangan aplikasi di PT Tabel Data Informatika secara menyeluruh. Dengan mengintegrasikan CI/CD dalam budaya dan proses pengembangan perusahaan, diharapkan PT Tabel Data Informatika dapat terus menjadi pemimpin dalam menyediakan solusi teknologi informasi yang inovatif dan handal bagi klien-klien mereka, serta dapat beradaptasi dengan tantangan dan perubahan yang terus berkembang dalam industri teknologi informasi.

II. METODOLOGI PENELITIAN

Identifikasi Kebutuhan dan Perancangan Aplikasi: Tahap awal dari penelitian ini adalah mengidentifikasi kebutuhan dalam pengembangan aplikasi E-RBA di PT Tabel Data Informatika. Penelitian ini akan mengumpulkan informasi tentang masalah-masalah yang dihadapi dalam proses deploy manual saat ini, serta mengidentifikasi manfaat dan tujuan

yang diharapkan dari penggunaan CI/CD, seperti peningkatan efisiensi, kualitas, dan daya saing aplikasi. Tahap ini juga akan mencakup perancangan dan konfigurasi sistem CI/CD menggunakan Gitlab CI untuk aplikasi E-RBA. Proses ini melibatkan pengaturan otomatisasi tahapan-tahapan dalam pengembangan, seperti integrasi kode, deployment otomatis ke server produksi, dan monitoring aplikasi yang sudah berjalan.

Pengembangan Program Otomatisasi dan Implementasi serta uji coba: Pada tahap ini, penelitian akan mengembangkan program otomatisasi menggunakan Gitlab CI sesuai dengan kebutuhan dan standar perusahaan PT Tabel Data Informatika. Program ini akan diintegrasikan dengan repositori kode proyek E-RBA, sehingga setiap perubahan pada kode akan mengaktifkan pipeline CI/CD dan mengotomatisasi proses pengembangan. Program otomatisasi yang dikembangkan akan diimplementasikan dan diuji coba pada proyek E-RBA di PT Tabel Data Informatika. Pengujian akan meliputi validasi keberhasilan deployment, pengujian integrasi, dan uji coba fungsional aplikasi untuk memastikan bahwa program otomatisasi berjalan sesuai dengan harapan dan memenuhi standar perusahaan.

Evaluasi dan Analisis: Tahap ini akan mengevaluasi hasil dari implementasi CI/CD pada proyek E-RBA. Evaluasi akan mencakup perbandingan kinerja sebelum dan setelah penerapan CI/CD, termasuk waktu deployment, kualitas perangkat lunak, dan efisiensi pengembangan. Analisis akan mengidentifikasi keberhasilan dan tantangan dari penggunaan CI/CD dalam proyek E-RBA.

Rekomendasi dan Kesimpulan: Berdasarkan hasil evaluasi dan analisis, penelitian ini akan menyusun rekomendasi mengenai penerapan CI/CD dalam pengembangan aplikasi E-RBA di PT Tabel Data Informatika. Rekomendasi ini akan mencakup langkah-langkah untuk meningkatkan efisiensi dan kualitas pengembangan, serta penerapan CI/CD pada proyek-proyek selanjutnya di perusahaan. Kesimpulan akan merangkum temuan dan manfaat dari penelitian ini terhadap peningkatan efisiensi dan kualitas pengembangan aplikasi di PT Tabel Data Informatika melalui CI/CD.

III. IMPLEMENTASI

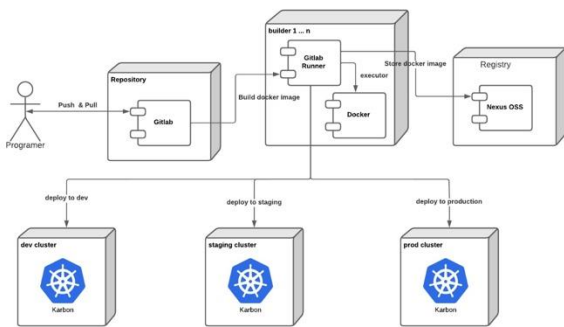
A. Gambaran Sistem Saat Ini dan Kebutuhan

Pada saat ini, aplikasi E-RBA di PT Tabel Data Informatika masih mengandalkan proses deploy tradisional yang melibatkan serangkaian langkah repetitif ketika terjadi pembaruan baik dalam skala major maupun minor. Proses deploy tersebut terdiri dari beberapa langkah, di antaranya programmer harus membuat paket distribusi dengan perintah "mvn clean package", yang kemudian menghasilkan file .jar untuk di-deploy di web server. Setelah itu, file .jar harus dipindahkan menggunakan protokol SFTP ke dalam server pengembangan, dan dilanjutkan dengan menjalankan file .jar menggunakan perintah "java -jar <nama-file>.jar". Proses tersebut juga mencakup langkah memastikan perubahan seperti konfigurasi database dan konfigurasi port, serta merestart systemd jika ada konfigurasi baru. Langkah-langkah yang repetitif ini menyebabkan proses deploy menjadi rentan terhadap kesalahan manusia dan memerlukan waktu yang cukup lama.

Untuk mengatasi masalah ini, diperlukan perubahan dalam proses deploy dengan memanfaatkan Continuous

Integration/Continuous Deployment (CI/CD). CI/CD akan memungkinkan otomatisasi dari tahap-tahap dalam pengembangan aplikasi, termasuk proses pengujian, integrasi kode, dan otomatisasi proses deploy ke server produksi. Dengan mengadopsi CI/CD, programmer tidak perlu lagi melakukan langkah-langkah manual yang repetitif seperti sebelumnya. Sebagai gantinya, setiap kali ada perubahan pada repositori kode, sistem CI/CD akan secara otomatis membangun paket distribusi, melakukan pengujian otomatis, dan mendeploy aplikasi ke server produksi dengan cepat dan konsisten. Dengan demikian, CI/CD akan meningkatkan efisiensi dan kualitas pengembangan aplikasi E-RBA di PT Tabel Data Informatika, serta mengurangi risiko kesalahan manusia dan penundaan rilis produk.

B. Pengembangan Program Otomatisasi dan implementasi serta uji coba



GAMBAR III.1 Alur pengembangan sistem

Pengembangan program otomatisasi untuk implementasi Continuous Integration/Continuous Deployment (CI/CD) pada aplikasi E-RBA di PT Tabel Data Informatika melibatkan serangkaian langkah yang cermat dan efisien. Alur pengembangan dimulai dengan partisipasi programmer yang memulai pekerjaannya dengan menarik kode dari Gitlab. Setelah selesai dengan pengembangan, programmer mengunggah kode hasil pekerjaannya kembali ke dalam Gitlab.

Kemudian, tanpa diketahui oleh programmer, Gitlab akan berinteraksi dengan Gitlab Runner melalui REST API untuk memulai proses otomatisasi. Gitlab Runner berfungsi sebagai executor yang bertugas menjalankan task yang sudah ditulis dalam format YAML, yang merepresentasikan tahapan-tahapan dalam proses CI/CD. Tahapan-tahapan ini mencakup pengujian otomatis, build kode, dan deploy aplikasi.

```
build:docker:
  stage: build
  extends: .build-docker
  variables:
    DOCKER_BUILD_ARGS_ENTRIPPOINT: "--build-arg JAR_FILE=$JAR_FILE"
  needs:
    - build:jar
    - get-fact:project:info
  rules:
    - if: $CI_COMMIT_TAG
      when: always
```

GAMBAR III.2 Skript GitlabCi – Build Docker

Selanjutnya, Gitlab Runner akan menggunakan Docker untuk membuat docker image dari aplikasi berdasarkan Dockerfile yang telah disediakan di dalam repository proyek. Docker image ini kemudian diunggah ke dalam Nexus OSS,

sebuah repositori Docker image yang aman dan terkelola dengan baik.

Terakhir, Gitlab Runner memerintahkan Kubernetes Master untuk menarik Docker image yang ada di Nexus OSS. Image ini akan dijalankan dalam sebuah pod di lingkungan Kubernetes, sehingga aplikasi E-RBA akan berjalan di dalam pod tersebut dengan cara yang terisolasi dan aman.

Dengan alur pengembangan yang terperinci dan otomatis ini, penggunaan CI/CD pada proyek E-RBA dapat meningkatkan efisiensi dan kualitas pengembangan secara signifikan. Setiap perubahan kode yang diunggah ke Gitlab akan memicu otomatisasi penuh mulai dari pembangunan aplikasi hingga pengimplementasian di lingkungan produksi, menghasilkan rilis yang cepat, konsisten, dan handal. Proses otomatis ini juga mengurangi risiko kesalahan manusia dan penundaan rilis, serta meningkatkan transparansi dan keterlibatan seluruh tim pengembang dalam proses pengembangan aplikasi E-RBA.

IV. EVALUASI DAN ANALISIS

Hasil dari penelitian ini merupakan sebuah prosedur deploy yang telah diotomatisasi untuk mengimplementasikan Continuous Integration/Continuous Deployment (CI/CD) pada aplikasi E-RBA di PT Tabel Data Informatika. Proses deploy telah dibagi menjadi pekerjaan-pekerjaan kecil yang mencakup tugas-tugas spesifik, dengan tujuan untuk memudahkan tim dalam melakukan investigasi ketika terjadi kesalahan atau masalah pada proses deploy.

```
ARG JDK_VERSION=11-oraclelinux8
ARG PRIVATE_REGISTRY_PULL=private.nexus-regs.docker:8086/

FROM ${PRIVATE_REGISTRY_PULL}openjdk:${JDK_VERSION}

RUN groupadd www-data 66 \
  adduser -r -g www-data www-data

WORKDIR /usr/local/share/applications
USER www-data

ENV APPLICATION_PORT=example
ENV PROFILE=example
ENV DATABASE_SID=example
ENV DATABASE_USER=example
ENV DATABASE_PASSWORD=example
ENV DATABASE_HOST=example
ENV DATABASE_NAME=example
ENV DATABASE_PORT=example
ENV FLYWAY_ENABLED=example

ARG JAR_FILE="saath2-server-oracle-2020.06.03.20.34-final.jar"
COPY --chown=www-data:www-data target/${JAR_FILE} spring-boot.jar

ENTRYPOINT ["java", "-jar", "-Djava.security.egd=file:/dev/./urandom", "spring-boot.jar"]

CMD ["--server.port=${APPLICATION_PORT}", "--spring.profiles.active=${PROFILE}"]

EXPOSE ${APPLICATION_PORT}/tcp

HEALTHCHECK --interval=5s --timeout=3s \
  CMD curl -f http://localhost:${APPLICATION_PORT}/actuator/health || exit 1
```

GAMBAR IV.1 Skrip Dockerfile

Pertama, luaran dari penelitian ini berupa Dockerfile, sebuah skrip instruksi yang berfungsi untuk membungkus satu service aplikasi. Dockerfile ini memungkinkan proses pembuatan docker image dari aplikasi E-RBA, yang nantinya akan dijalankan di dalam lingkungan Docker untuk memastikan portabilitas dan isolasi aplikasi.

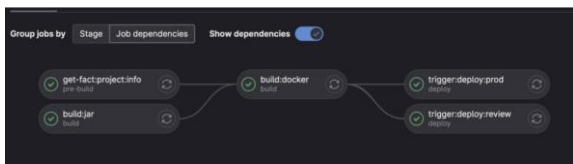
```

build-jar:
  stage: build
  extends: .build-jar
  rules:
    - if: $CI_COMMIT_TAG
      when: always
  build-docker:
    stage: build
    extends: .build-docker
    variables:
      DOCKER_BUILD_ARGS_ENTRYPOINT: "--build-arg JAR_FILE=$JAR_FILE"
    needs:
      - build-jar
    - get-fact:project:info
    rules:
      - if: $CI_COMMIT_TAG
        when: always
  trigger:deploy:review:
    extends: .trigger:deploy:review
    stage: deploy
    needs:
      - get-fact:project:info
      - build-docker
    rules:
      - if: $CI_ENV_REVIEW_ENABLED == "true" && $CI_COMMIT_TAG
        when: always
      - if: $CI_COMMIT_REF_SLUG == $CI_DEFAULT_BRANCH && $CI_ENV_REVIEW_ENABLED == "true" && $CI_COMMIT_TAG
        when: never

```

GAMBAR IV.2
Skrip GitlabCI

Kedua, hasil penelitian ini mencakup skrip Gitlab CI yang berisi daftar task yang akan dijalankan saat melakukan proses deploy aplikasi ke lingkungan server. Skrip Gitlab CI ini ditulis dalam format YAML agar dapat dibaca dan dieksekusi oleh Gitlab Runner. Task-task dalam Gitlab CI mencakup langkah-langkah pengujian otomatis, build kode, dan otomatisasi proses deploy ke server produksi.



GAMBAR IV.3
Tahapan alur perintah

Ketiga, penelitian ini menghasilkan tahapan-tahapan prosedur yang telah diotomatisasi dalam CI/CD. Tahapan-tahapan ini mencakup pembuatan docker image menggunakan Dockerfile yang ada di dalam repository, mengunggah docker image ke dalam Nexus OSS, serta memerintahkan Kubernetes Master untuk menarik docker image dari Nexus OSS dan menjalankannya di dalam pod Kubernetes.

Evaluasi dan analisis dari hasil penelitian ini menunjukkan bahwa implementasi Continuous Integration/Continuous Deployment (CI/CD) pada aplikasi E-RBA di PT Tabel Data Informatika memberikan manfaat yang signifikan bagi perusahaan. Dengan adanya prosedur deploy yang diotomatisasi melalui Dockerfile dan skrip Gitlab CI, tim pengembang mengalami peningkatan efisiensi dalam proses pengembangan dan pengimplementasian aplikasi. Proses deploy yang sebelumnya cenderung repetitif dan memakan waktu, kini menjadi lebih efisien dan cepat berkat otomatisasi yang dilakukan.

Penggunaan Dockerfile memungkinkan pembuatan docker image yang dapat digunakan untuk menjalankan aplikasi E-RBA di lingkungan Docker, sehingga memastikan portabilitas dan isolasi aplikasi. Selain itu, skrip Gitlab CI dalam format YAML membantu tim dalam menjalankan tugas-tugas deploy secara terstruktur dan terdokumentasi dengan baik, sehingga memudahkan identifikasi dan penyelesaian masalah yang mungkin terjadi selama proses deploy.

Dengan adanya CI/CD, proses pengembangan aplikasi E-RBA menjadi lebih konsisten dan handal, mengurangi risiko kesalahan manusia dan memastikan bahwa setiap perubahan kode yang diunggah ke Gitlab dapat segera diintegrasikan dan di-deploy secara otomatis. Hal ini berkontribusi dalam

meningkatkan kualitas dan stabilitas aplikasi yang dirilis ke lingkungan produksi.

Selain manfaat efisiensi dan kualitas, implementasi CI/CD juga memberikan dampak positif dalam meningkatkan transparansi dan keterlibatan seluruh tim pengembang dalam proses pengembangan aplikasi. Setiap langkah dalam proses CI/CD terdokumentasi secara rinci, sehingga setiap anggota tim dapat dengan mudah memahami tahapan-tahapan yang dilakukan dalam deploy aplikasi.

Namun, evaluasi ini juga mengindikasikan beberapa tantangan yang mungkin dihadapi selama implementasi CI/CD. Penerapan CI/CD memerlukan adopsi budaya DevOps di perusahaan, yang melibatkan kolaborasi antara tim pengembang, tim operasi, dan pihak-pihak terkait lainnya. Selain itu, diperlukan investasi dalam sumber daya dan infrastruktur teknologi guna mendukung keberhasilan CI/CD.

Secara keseluruhan, hasil penelitian ini memberikan kontribusi positif dalam meningkatkan efisiensi dan kualitas pengembangan aplikasi E-RBA di PT Tabel Data Informatika. Dengan CI/CD, perusahaan dapat lebih siap menghadapi tantangan dalam era teknologi informasi yang terus berkembang pesat, serta dapat memberikan solusi teknologi informasi yang lebih baik dan lebih cepat kepada klien-klien mereka.

V. KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan penelitian ini, implementasi otomatisasi deployment menggunakan Continuous Integration/Continuous Deployment (CI/CD) telah memberikan manfaat yang signifikan bagi PT Tabel Data Informatika. Dengan adanya prosedur deploy yang diotomatisasi, tim pengembang dapat melakukan proses deployment aplikasi E-RBA secara lebih cepat dan efisien. Proses yang sebelumnya memakan waktu dan rawan kesalahan manusia dapat dihindari dengan otomatisasi ini. Penggunaan Dockerfile dan skrip Gitlab CI memungkinkan langkah-langkah dalam proses deploy menjadi terstruktur dan terdokumentasi dengan baik, sehingga memudahkan tim dalam mengidentifikasi dan menyelesaikan masalah yang mungkin timbul selama proses deploy.

Selain itu, implementasi CI/CD juga telah meningkatkan kualitas dan stabilitas aplikasi E-RBA yang dirilis ke lingkungan produksi. Setiap perubahan kode yang diunggah ke Gitlab dapat segera diintegrasikan dan di-deploy secara otomatis, sehingga memastikan bahwa aplikasi yang dirilis adalah versi terbaru dan sesuai dengan standar perusahaan. Proses deploy yang konsisten dan handal juga telah memberikan kontribusi positif dalam meningkatkan kepercayaan klien terhadap aplikasi E-RBA.

B. Saran

Sebagai saran untuk pengembangan E-RBA di masa mendatang, disarankan untuk melakukan unit testing secara otomatis setiap kali melakukan deploy aplikasi. Dengan melakukan unit testing secara otomatis, tim pengembang dapat dengan cepat mendeteksi dan memperbaiki masalah atau bug yang mungkin muncul setelah deployment. Unit testing yang terintegrasi dalam proses CI/CD akan memastikan bahwa setiap perubahan kode yang diunggah telah melewati serangkaian pengujian yang ketat sebelum

diterapkan di lingkungan produksi. Hal ini akan membantu mengurangi risiko terhadap potensi kesalahan dan masalah yang dapat mempengaruhi kinerja dan kehandalan aplikasi.

Selain itu, perusahaan sebaiknya terus mendorong budaya kolaborasi antara tim pengembang dan tim operasi (DevOps). Kolaborasi yang erat antara kedua tim ini akan memfasilitasi proses implementasi CI/CD secara lebih lancar. Selain itu, perusahaan juga harus berinvestasi dalam infrastruktur teknologi yang memadai untuk mendukung keberhasilan implementasi CI/CD. Ketersediaan sumber daya dan teknologi yang memadai akan menjadi kunci kesuksesan dalam mencapai efisiensi dan kualitas yang diharapkan dari proses CI/CD.

Dengan mengambil langkah-langkah di atas, PT Tabel Data Informatika dapat terus meningkatkan efisiensi dan kualitas pengembangan aplikasi E-RBA, serta meningkatkan daya saing di pasar yang kompetitif. Implementasi CI/CD yang baik akan membantu perusahaan tetap relevan dalam era teknologi informasi yang terus berkembang pesat dan memberikan solusi teknologi informasi yang handal dan inovatif kepada klien-klien PT Tabel Data Informatika itu sendiri.

REFERENSI

- [1] GitLab. "What is Continuous Integration (CI)?" [Online]. Tersedia: <https://docs.gitlab.com/ee/ci/introduction/> [Diakses pada 23 Juli 2023].
- [2] Gilang V. Perdana, "Monitoring Kubernetes Cluster Bersama Lens", [online]. Tersedia: <https://gbesar.com/main/baca/Monitoring%20Kubernetes%20Cluster%20bersama%20Lens/all> [Diakses pada 23 Juli 2023].
- [3] Atlassian. "What is Continuous Deployment?" [Online]. Tersedia: <https://www.atlassian.com/continuous-delivery/continuous-deployment> [Diakses pada 23 Juli 2023].
- [4] Fowler, M. "Continuous Integration." [Online]. Tersedia: <https://martinfowler.com/articles/continuousIntegration.html> [Diakses pada 23 Juli 2023].
- [5] Docker. "Docker Overview." [Online]. Tersedia: <https://www.docker.com/what-docker> [Diakses pada 23 Juli 2023].
- [6] GitLab. "Continuous Integration." [Online]. Tersedia: <https://docs.gitlab.com/ee/ci/> [Diakses pada 23 Juli 2023].
- [7] Shah, A. "Kubernetes Tutorial: Learn Kubernetes Step-by-Step." Guru99, 2022. [Online]. Tersedia: <https://www.guru99.com/kubernetes-tutorial.html> [Diakses pada 23 Juli 2023].