

## Pengembangan dan Pembaruan Manajemen Aplikasi pada Server

<sup>1</sup>Agus Fatulloh, <sup>2</sup>Alfatha Dhiki, <sup>3</sup>Noranizah, <sup>4</sup>Eka Wahyuni

<sup>1,3,4</sup>Teknik Informatika Politeknik Negeri Batam, <sup>2</sup>Dinas KomInfo Kota Batam

Jl. Ahmad Yani, Tlk. Tering, Kota Batam, Kepulauan Riau

<sup>1</sup>agusf@polibatam.ac.id, <sup>2</sup>alfatha92@gmail.com, <sup>3</sup>noranizah11@gmail.com, <sup>4</sup>1997ekawahyuni@gmail.com

---

### ABSTRACT

In KomInfo Office, there is a Developer team that is tasked with developing applications. They save project on GitLab so that every member gets the latest. This project must also be stored on the server by Administrator. The Administrator must log in first to enter the server then pull the project files from GitLab and then save it on the server. This makes the project development process take longer, because it is necessary to develop a system that can connect GitLab and the server using the Microservices file on the server with the aim that the Administrator or Developer can directly save files to the server without need to log in repeatedly. The system development uses the spiral method so that in case of changes, additions, or system development can be done quickly according to the user's wishes. System testing is carried out using the black box method to ensure all functions are running normally. Measurement of the effectiveness of the system is carried out by survey and calculated using a Likert scale. As a result, this system was able to increase the effectiveness of work, especially those related to coordination, access, management, and speed of project completion.

---

*Keywords* - Administrator, GitLab, Microservices, Projek, Server.

---

### ABSTRAK

Di Dinas KomInfo terdapat tim Developer yang bertugas untuk melakukan pembuatan, pengembangan, dan pemeliharaan aplikasi. Mereka menyimpan hasil pembaruan proyek di GitLab agar setiap anggota mendapatkan update terbaru dari proyek yang sedang dikerjakan. Selain di simpan di GitLab File proyek ini juga harus disimpan di server dan yang melakukannya harus seorang Administrator. Administrator harus login terlebih dahulu untuk masuk ke server kemudian pull file proyek dari GitLab dan setelah itu baru disimpan di Server. Hal ini menjadikan proses pengembangan proyek membutuhkan waktu yang lebih lama karena harus menyelesaikan kedua tahap tersebut, karena itu dibutuhkanlah pengembangan sistem yang dapat menghubungkan antara GitLab dengan server menggunakan file Microservices yang diletakkan di server dengan tujuan agar Administrator ataupun Developer dapat langsung menyimpan file ke server tanpa perlu login berulang kali ke server tetapi cukup melakukan aksi deploy di GitLab. Metode pengembangan sistem menggunakan metode spiral agar jika terjadi perubahan, penambahan, atau pengembangan sistem dapat dilakukan secara cepat menurut keinginan user. Pengujian sistem dilakukan dengan metode black box untuk memastikan seluruh fungsi yang ada berjalan normal. Pengukuran efektifitas sistem dilakukan dengan survey dan dihitung menggunakan skala likert. Hasilnya sistem ini mampu meningkatkan efektifitas pekerjaan khususnya yang terkait dengan koordinasi, akses, pengelolaan, dan kecepatan penyelesaian proyek.

---

*Kata Kunci* - Administratror, GitLab, Microservices, Projek, Server.

---

## 1. Introduction

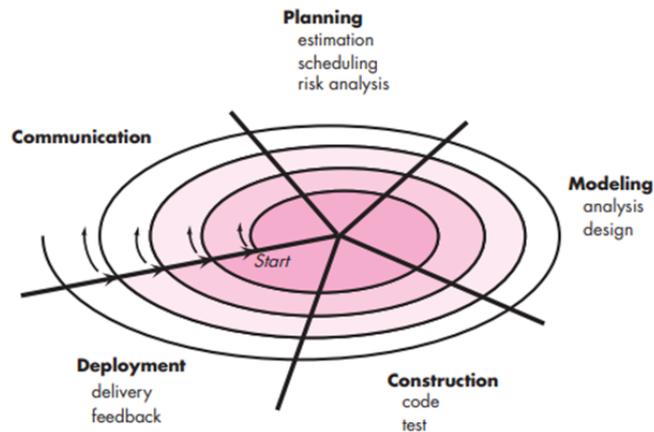
Dinas KomInfo Kota Batam memiliki tim Developer yang bertugas untuk melakukan pembuatan, pengembangan dan pemeliharaan aplikasi serta Administrator yang akan memasukkan file proyek dari Developer ke dalam server. Dalam hal ini dibutuhkan komunikasi antara Administrator dengan Developer untuk mendapatkan file proyek dan proses ini memakan waktu apalagi jika dari ke dua belah pihak tidak segera merespon permintaan yang ada. Penggunaan GitLab sebagai media untuk menyimpan file proyek secara *cloud* dari Developer tentu saja dapat mempermudah dalam hal pengelolaan pengembangan proyek nya, namun pengembangan proyek yang memerlukan komunikasi dua arah dari dua pihak dalam hal ini antara Developer dengan Administrator di Gitlab belum maksimal, karena komunikasi nya masih secara manual seperti melalui aplikasi pengirim pesan yang itu bukan dilakukan secara real time ketika ada file proyek yang telah selesai dikerjakan, sehingga membutuhkan waktu yang lebih panjang dan Administrator pun harus berulang kali melakukan log in ke server untuk melakukan penyimpanan file proyek nya ke server tersebut. Faktor ini sering menyebabkan terjadinya kesalahan atau terdapat file ganda sehingga terjadi konflik file.

Berdasarkan permasalahan di atas maka diperlukan sebuah sistem yang dapat menghubungkan antara GitLab dengan server yang ada, sehingga pihak Developer maupun Administrator dapat langsung melakukan penyimpanan hasil update file proyeknya ke server secara langsung yang disebut dengan sistem *microservice* yang tentu saja tetap dalam pengawasan Administrator. Beberapa fitur yang ada pada GitLab selain mempermudah dalam hal pengelolaan proyek, Administrator juga akan dapat melihat apa saja yang dikerjakan oleh Developer dan siapa yang mengerjakannya, serta dapat mengetahui jika file yang dikerjakan terdapat error. Selain itu, sistem ini juga dilengkapi dengan adanya notifikasi ke Administrator menggunakan bot telegram ketika dilakukan push file ke GitLab.

## 2. Research Method

Metode penelitian menggunakan metode model spiral yang terdiri dari beberapa tahapan, yang pertama adalah *communication* (Komunikasi), pada tahap ini melakukan wawancara dan diskusi dengan Sistem Administrator terhadap permasalahan yang ada dan kebutuhan fungsional dari sistem yang akan dikembangkan. Tahap kedua adalah *planning* (perencanaan), pada tahapan ini dilakukan perencanaan untuk pengembangan sistem berdasarkan informasi yang didapat saat wawancara, sekaligus perkiraan resiko yang mungkin terjadi. Perencanaan yang dilakukan adalah membuat sistem yang dapat menghubungkan secara otomatis antara GitLab dengan server yang nantinya baik Developer maupun Administrator dapat menyimpan serta berbagi file proyek aplikasi dengan anggota lainnya dan Administrator juga dapat melihat aktivitas pengerjaan proyek serta akan mendapatkan notifikasi bot telegram jika ada push file ke GitLab. Tahap ketiga *modelling* (pemodelan), pada tahap ini membuat rancangan dari alur kerja sistem yang akan digunakan, analisis terhadap penyelesaian masalah, membuat algoritma atau pseudocodenya. Tahap keempat adalah *construction* (Konstruksi), pada tahap ini adalah pembuatan program *microservices* untuk menjembatani aplikasi GitLab dengan Server yang nantinya Administrator ataupun Developer dapat melakukan penyimpanan file proyek ke server dengan pantauan Administrator. Tahap kelima *Deployment* (penyebaran), pada tahap ini dilakukan penyerahan produk yang telah dikembangkan sekaligus untuk mendapatkan masukan dari user mengenai produknya.

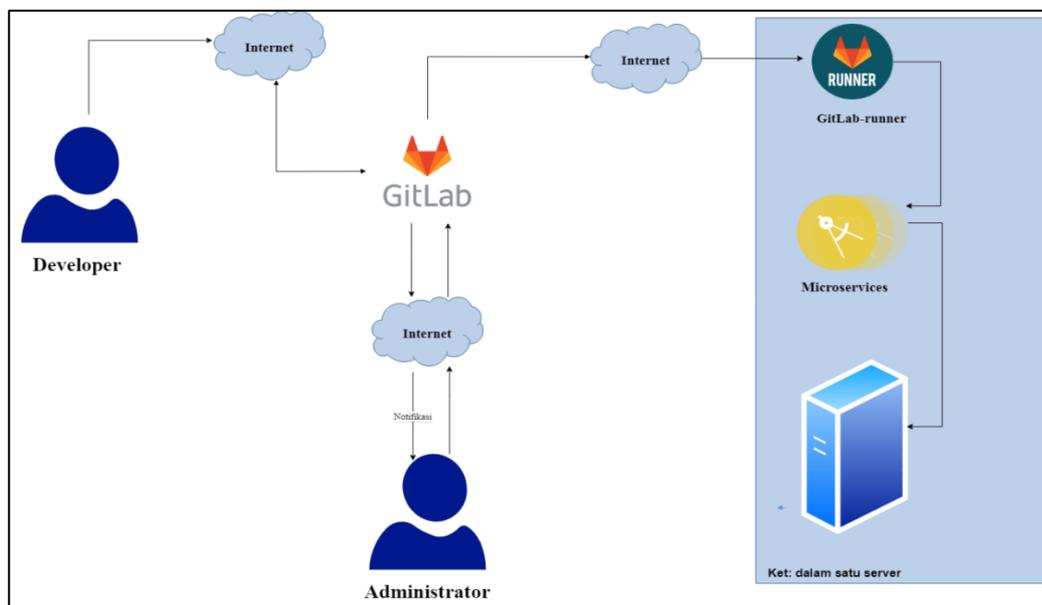
Perencanaan di antaranya adalah membuat penjadwalan untuk penyelesaian produk yang diinginkan sampai pada waktu penyerahan produk. Penjadwalan yang disusun seperti terlihat pada tabel 1. Berdasarkan hasil wawancara penggambaran model interaksi user dengan sistem yang akan dikembangkan seperti terlihat pada gambar 2.



Gambar 1. Model Spiral [ sumber: Annakota, 2018 ]

Tabel 1. Penjadwalan Penyelesaian Pengembangan Sistem

No	Kegiatan	Minggu Ke-													
		1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	Wawancara	█													
2	Perencanaan		█												
3	Analisis			█											
4	Konstruksi dan Pengujian				█										
5	Penyerahan Sistem														█

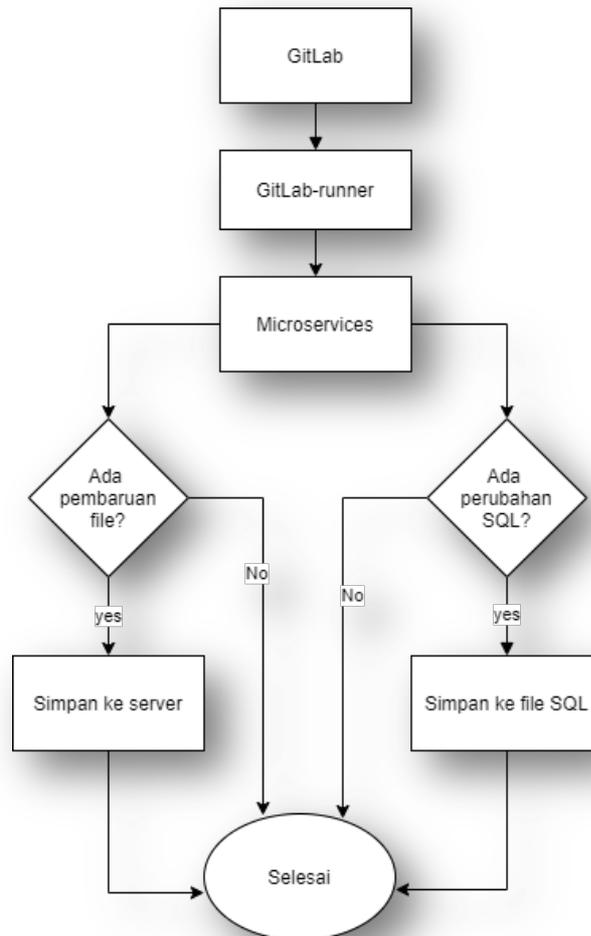


Gambar 2. Deskripsi Umum Sistem

Pihak Developer melakukan pengembangan proyeknya melalui sebuah repositori GitLab yang sekaligus menyimpan hasilnya di sana dan pihak Administrator akan mendapatkan notifikasi jika ada penambahan atau pembaruan file proyek yang dilakukan oleh tim Developer. File proyek yang ada di GitLab selanjutnya akan di *Pull* ke server melalui GitLab-Runner menggunakan microservices,

sehingga pada saat yang sama file proyek akan tersimpan secara otomatis ke server. Pihak Administrator juga akan dapat memantau dan melihat log aktivitas pembaruan yang terjadi di GitLab.

Alur kerja dari implementasi *microservices* seperti terlihat pada gambar 3 yaitu terdapat dua layanan. Pertama adalah layanan jika ada pembaruan file proyek di GitLab, maka GitLab-runner akan melakukan pull file proyek menggunakan *microservices* yang selanjutnya akan dilakukan penyimpanan ke server. Kedua adalah jika ada pembaruan file SQL di GitLab, maka GitLab-runner akan melakukan pull file SQL menggunakan *microservices* maka file SQL akan langsung ter-update sesuai dengan perubahan yang dilakukan.



**Gambar 3.** Flowchart Microservices Pada Manajemen Aplikasi

### 3. Result and Analysis

#### 3.1. Konstruksi dan Pengujian

Peran *microservices* adalah untuk memudahkan baik pihak Developer maupun Administrator dalam melakukan penyimpanan file ke server secara langsung ketika adanya penambahan atau perubahan file proyek di GitLab atau ketika melakukan update script SQL. Ada dua layanan yang dilakukan oleh *microservices*, yaitu ketika terjadi pembaruan file proyek ataupun perubahan script SQL. Ada pun pseudocode untuk layanan pembaruan file proyek seperti berikut:

```
Pesan = 'menjalankan cloningan file dari gitlab'  
copy file: copy file project dari <folder gitlab> ke <folder server>
```

copy file: copy file koneksi di server dari <folder asal> ke <folder tujuan>  
tampil isi Pesan

dan pseudocode untuk layanan perubahan script SQL seperti berikut:

```
getcwd()
menampilkan pesan 'menjalankan SQL'
try {
  $check_file = adakah file sql?
  if( ada )
    $sqls = sql-to-execute
    if( $sqls != "" )
      sqls <- mengambil isi <file eksekusi sql>
      foreach ($sqls as $value)
        tampil isi $value
        mengambil isi <file query sql>
        lakukan koneksi ke database
        simpan $value ke database
        tampil pesan "melakukan eksekusi file sql:" $value
      else
        tampil pesan "file tidak ditemukan"
    }
}
```

Pembaruan yang terjadi akan diinformasikan menggunakan teknologi Bot Telegram yang disisipkan dalam suatu grup telegram yang berisi Developer dan Administrator. *Script* ini disematkan di dalam setiap program microservices. Pada script Telegram ini berisi token bot telegram dan id grup telegram. Berikut *script* telegram yang akan digunakan untuk mengirim notifikasi.

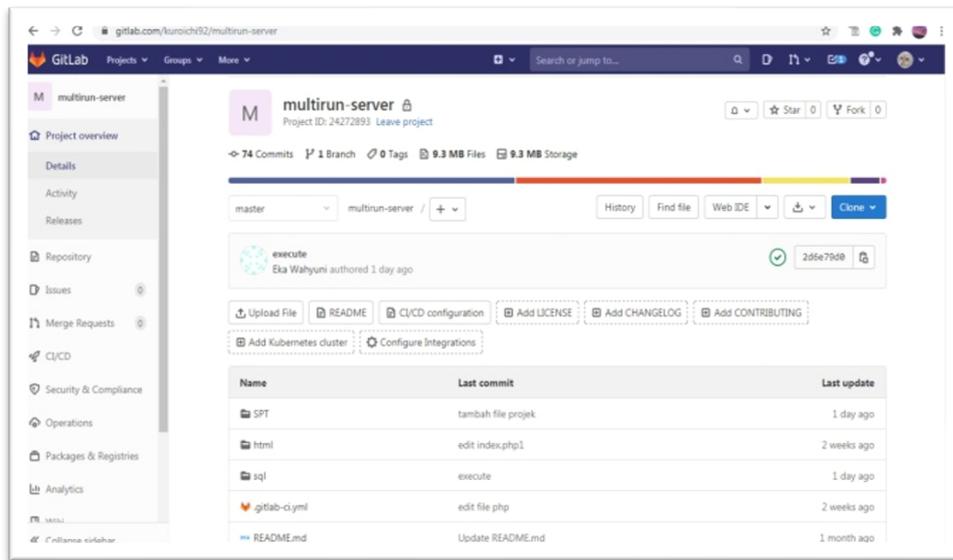
```
#telegram
$var tokentele = 'xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx'; // Token Bot Tele
$var id grup = 'yyyyyy'; // ID Groups Tele
file_get_contents('https://api.telegram.org' . $var
tokentele . '/sendMessage?chat_id=')
```

### 3.2 Penyerahan Produk

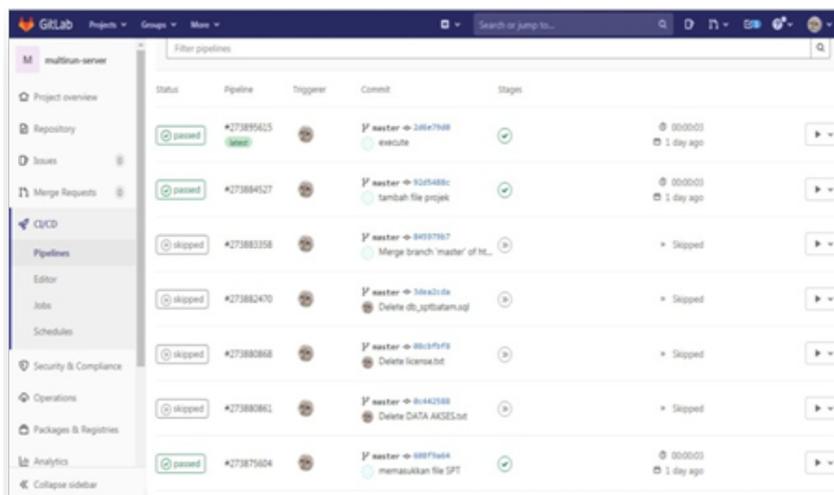
#### 1. Implementasi

Dalam implementasi microservices terdiri dari beberapa tahapan, tahapan pertama tim Developer harus melakukan penyimpanan file proyek ke GitLab dengan menjalankan beberapa perintah di shell. Perintah yang pertama `git add *`. Perintah ini sebagai persiapan awal untuk menambahkan file baru atau file perubahan di lokal yang akan digunakan di GitLab. Perintah yang kedua `git commit -am "isi pesan commit"`. Perintah ini untuk melakukan penyimpanan perubahan ke repositori menggunakan opsi `-am` yang artinya shell akan menginformasikan secara otomatis jika ada file yang diubah atau dihapus dengan menambahkan pesan tertentu dari user. Perintah yang ketiga `git push`. Perintah ini untuk melakukan penyimpanan semua perubahan yang ada di lokal commit atau memasukkannya ke remote repositori. Hasil push file seperti terlihat pada Gambar 4 dimana file SPT merupakan contoh proyek.

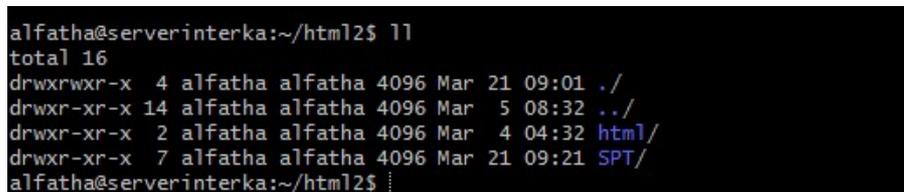
Tahapan yang kedua yaitu ketika file proyek sudah ada di repositori GitLab, ketika pihak Developer melakukan pembaruan atau update file proyeknya dan melakukan perubahan status file tersebut yang berada di menu CI/CD  pipelines lalu menyatakan passed maka secara otomatis file proyek tersebut akan disimpan ke dalam server dan secara otomatis memberikan notifikasi ke grup telegram. Gambar 5 menunjukkan halaman pipelines GitLab, gambar 6 hasil penyimpanan otomatis ketika status passed diberikan pada file proyeknya, dan gambar 7 notifikasi pada telegram



Gambar 4. Hasil Push File Projek SPT Di GitLab



Gambar 5. Halaman Pipelines GitLab



Gambar 6. Tampilan Apabila Terjadi Perubahan Status Passed



Gambar 7. Notifikasi Pada Telegram

2. Pengujian

Pengujian implementasi microservices menggunakan metode uji black box untuk memeriksa fungsional secara lengkap. Hasil pengujian seperti tersaji pada tabel 2.

**Tabel 2.** Pengujian Implementasi Microservices

No	Uji Fungsi	Skenario Pengujian	Target	Hasil
1.	Push file ke GitLab	Menyediakan file yang akan di-push ke GitLab	File dapat tersimpan di GitLab	✓
2.	Copy file dari GitLab ke Server	Mengubah status file menjadi <i>passed</i> di GitLab	File proyek yang dinyatakan <i>passed</i> di GitLab akan terunduh otomatis ke server	✓
3.	Eksekusi query ke database	Melakukan update file SQL di GitLab	File SQL yang berada di server akan ter-update secara otomatis berdasarkan list “sql-to-execute” pada database tujuan	✓
4.	Notifikasi telegram	Melakukan push file dan dijalankan pada pipeline dengan memberikan status <i>passed</i>	Muncul notifikasi informasi pada grup telegram	✓

Dari hasil pengujian yang telah dilakukan semua fungsinya dapat berjalan dengan normal. Pengujian ini dilakukan oleh pihak Developer, Administrator, serta Pengembang sistem.

3. Kuesioner

Kuesioner diperlukan untuk mengukur seberapa besar manfaat dari sistem yang telah dikembangkan terhadap pengembangan proyek dan kolaborasi tim. Kuesioner disebar khusus di lingkungan Unit Developer Dinas KomInfo Kota Batam yang berjumlah 5 orang. Analisis data dari hasil rekapitulasi kuesioner menggunakan analisis skala likert. Daftar pertanyaan dan respon dari responden tersaji pada tabel 3, adapun untuk kriteria interpretasi intervalnya tersaji pada tabel 4.

**Tabel 3.** Hasil Kuesioner Implementasi Microservices

No	Pertanyaan	SS	S	N	TS	STS
1.	Apakah dengan adanya sistem ini dapat membantu Anda memperoleh informasi yang lebih cepat ketika terjadi perubahan data pada file proyek?	2	3	0	0	0
2.	Apakah dengan menggunakan sistem ini koordinasi dan kolaborasi pengerjaan proyek menjadi lebih mudah?	2	3	0	0	0
3.	Apakah dengan sistem ini membuat Anda menjadi lebih mudah dalam mengakses file proyek?	1	3	1	0	0
4.	Apakah dengan sistem ini pengelolaan file proyek menjadi lebih mudah?	2	2	1	0	0
5.	Apakah dengan sistem ini pengerjaan dan penyelesaian sebuah proyek menjadi lebih cepat?	1	3	1	0	0

**Tabel 4.** Kriteria Interpretasi Interval

No	Pilihan Jawaban	Interval Persentase
1.	Sangat Setuju (SS)	80% s.d 100%
2.	Setuju (S)	60% s.d 79.99%
3.	Netral (N)	40% s.d 59.99%
4.	Tidak Setuju (TS)	20% s.d 39.99%
5.	Sangat Tidak Setuju (STS)	0% s.d 19.99%

Adapun bobot jawaban yang diberikan telah ditentukan seperti terlihat pada tabel 5.

**Tabel 5.** Bobot Jawaban Kuesioner

No	Pilihan Jawaban	Interval Persentase
1.	Sangat Setuju (SS)	5
2.	Setuju (S)	4

3.	Netral (N)	3
4.	Tidak Setuju (TS)	2
5.	Sangat Tidak Setuju (STS)	1

Penentuan indeks[2] dalam bentuk persentase dengan rumus:

$$\text{Indeks (\%)} = \left( \frac{\text{Total Skor}}{\text{Skor Maksimum}} \right) \times 100 \quad (1)$$

Hasil analisis perhitungan kuesioner dengan skala likert tersaji pada tabel 6.

**Tabel 6.** Rekapitulasi Hasil Kuesioner Menggunakan Skala Likert

Pertanyaan No	Nilai Skor					Total Skor	Indeks
	SS	S	N	TS	STS		
1.	10	12	0	0	0	22	88%
2.	10	12	0	0	0	22	88%
3.	5	12	3	0	0	20	80%
4.	10	8	3	0	0	21	84%
5.	5	12	3	0	0	20	80%

Berdasarkan hasil rekapitulasi dari tabel 6 terlihat bahwa indeks persentase dari semua pertanyaan lebih besar sama dengan 80% yang artinya jika mengacu pada tabel 4 dapat dinyatakan bahwa pihak Developer semuanya menyatakan sangat setuju dengan implementasi microservices ini.

#### 4. Conclusion

Dari seluruh rangkaian pengembangan dan pengujian sistem dapat disimpulkan bahwa implementasi microservices mempermudah tim Developer dan Administrator ketika ingin menyimpan file projek dari GitLab ke server, koordinasi dan komunikasi menjadi lebih mudah dan cepat karena adanya notifikasi ke grup melalui telegram, dan pihak Administrator pun dapat memantau projek yang sedang dikerjakan oleh tim Developer secara langsung.

#### References

- [1] Anakotta, G. L., & Adhy, S. (2018). Perancangan dan Implementasi E Commerce Dengan Segmentasi Harga Menggunakan Metode Pengembangan Spiral (Studi Kasus : CV. Citra Mandiri Bandarlampung). Jurnal Masyarakat Informatika
- [2] Athailah. (2011). Mastering Ubuntu.
- [3] Jakarta Selatan: MediaKita.
- [4] Fazil, & Hendrawaty. (2019). Rancang Bangun Sistem Inventaris Barang Berbasis Web Dengan Pemanfaatan Bot. Lhokseumawe, Aceh: Politeknik Negeri Lhokseumawe.
- [5] GitLab. (2011). GitLab Docs. Retrieved Oktober 15, 2020, from GitLab.com: <https://docs.gitlab.com/ee/ci/introduction/index.html#introduction-to-cicd-methodologies>
- [6] Mufrizal, R., & Indarti, D. (2019). Refactoring Arsitektur Microservice Pada Aplikasi Absensi PT. Graha Usaha Teknik. Jurnal Nasional Teknologi dan Sistem Informasi.
- [7] Nugroho, B. (2004). Aplikasi Pemrograman Web Dinamis dengan PHP dan. Yogyakarta: Gava Media.
- [8] Saputro, A. A., & Fatulloh, A. (2020). Motion Graphic Pada Video Mekanisme Permohonan Informasi Publik Di Badan Pengusaha Batam. Jurnal Teknologi Terapan.
- [9] Siregar, M. E. (2019). Implementasi Git Dan Github Untuk Membangun Aplikasi Menggunakan Android Studio.
- [10] Utomo, A. N., & Alfaridzi, M. (2018). Perancangan Sistem Informasi Pada Percetakan CV Citra Kencana Jakarta Timur Berbasis Web. Jurnal Rekayasa Informasi.
- [11] Wahanani, H. E., Saputra, W. S., & Wahono, B. V. (2019). Perancangan Infrastruktur Server Vcs (Version Control System) Dengan Gitlab Berbasis Git.