

DESIGN FRAMEWORK APLIKASI MOBILE BERBASIS DATA CLOUD COMPUTING

Leorimba Napitu

Sistem Informasi, STMIK Profesional Makassar

email: leorimba.napitu.09174141@gmail.com

Abstract

Applications in the latest generation of mobile devices are still constrained by power consumption, computing speed, memory size, wireless network bandwidth. Today, the popularity of smart devices and mobile networks has substantially changed the way people access computers and network services. To control the deployment and execution of applications, and what users need is a trusted service provider to host the virtual environment, which is much more practical than verifying more and more service providers. Cloud computing answers IT problems and challenges, including the problem of high IT investment budgets and disaster recovery plans as part of business continuity. Cloud computing consists of several layers, where each layer functions to support data communication between each service. Docker runs on the kernel layer, making it possible to run more virtualization with limited resources. Cloud computing systems on mobile devices can be implemented using the docker connect between mobile devices and cloud servers.

Kata Kunci : *Cloudcomputing, Docker, Mobile*

A. PENDAHULUAN

Teknologi komputasi seluler dan cloud telah memungkinkan tercipta aplikasi yang canggih. Namun, aplikasi di generasi terbaru perangkat seluler masih terkendala oleh konsumsi daya, kecepatan komputasi, ukuran memori, bandwidth jaringan nirkabel. Sejak Internet menjadi populer, perangkat seluler mungkin mengatasi kendala dengan mengatasi sebagian beban kerja aplikasi ke mesin server melalui jaringan untuk menghemat energi. Baru-baru ini, komputasi awan telah mengubah infrastruktur perangkat lunak dan bisnis model layanan Internet dengan teknologi untuk menyediakan dan mengelola sumber daya komputasi dan penyimpanan data yang melimpah melalui jaringan dengan biaya operasi diamortisasi yang relatif rendah[1]. Saat ini, popularitas perangkat pintar dan jaringan seluler telah secara substansial mengubah cara orang mengakses komputer dan layanan jaringan. Sedangkan kombinasi cloud computing dan mobile computing, disebut mobile cloud computing telah mulai menunjukkan efeknya dengan banyak aplikasi ponsel cerdas yang tampak lebih inovatif.

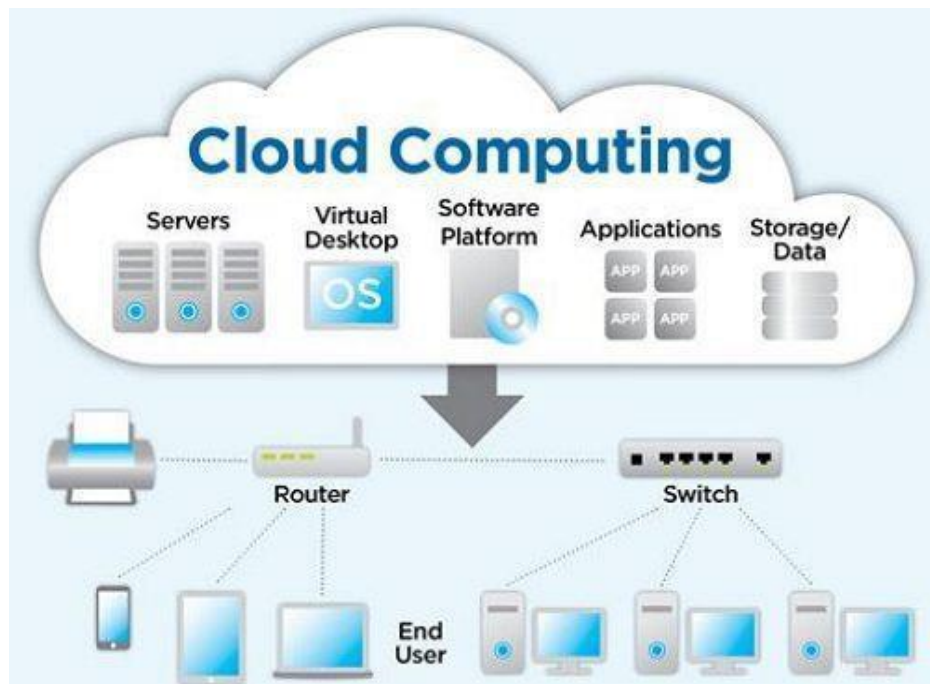
Berdasarkan permasalahan diatas, diusulkan kerangka kerja bagi pengguna untuk membuat lingkungan eksekusi virtual di cloud untuk menjalankan aplikasi seluler. Tidak seperti model client-server, desain ulang aplikasi tidak diperlukan oleh pengguna untuk dapat menjalankan aplikasi yang sudah ada pada perangkat fisik atau pada lingkungan virtual. Pendekatan ini memungkinkan pengguna untuk mengontrol penyebaran dan pelaksanaan aplikasi, dan apa yang dibutuhkan pengguna adalah penyedia layanan yang dapat dipercaya untuk menghosting lingkungan virtual, yang jauh lebih praktis daripada memverifikasi semakin banyak penyedia layanan. Selain menurunkan beban kerja dari lingkungan fisik, lingkungan virtual memberikan peluang untuk meningkatkan

fungsionalitas lingkungan eksekusi. Setelah mengurai permasalahan yang ditemukan, maka penulis mengusulkan ide penelitian dengan tema “Design Framework Apikasi Mobile Berbasis Data Cloud Computing”. Diharapkan penelitian ini dapat mendukung setiap perusahaan dalam meningkatkan dukungan layanan berbasis digital, mencakup latar belakang atas isu atau permasalahan serta urgensi dan rasionalisasi kegiatan (penelitian atau pengabdian).

B. METODE PENELITIAN

1. Framework Cloud Computing Mobile.

Komputasi awan menjawab masalah dan tantangan IT antaranya adalah masalah tingginya anggaran investasi IT dan rencana pemulihan bencana sebagai bagian dari business continuity[2]. Permasalahan tersebut dapat terjawab dengan baik melalui penerapan komputasi awan. Masalah lainnya, seperti tingginya tuntutan kebutuhan perusahaan, dapat terjawab dengan baik oleh komputasi awan dengan cara ketangkasan dalam pengembangan.



Gambar 1. Framework Cloud Computing Mobile

Teknologi cloud computing akan mudah diterapkan jika adanya kerangka kerja implementasi[4]. Sudah banyak kalangan industri, bisnis maupun institusi pemerintahan yang mengadopsi teknologi cloud, namun institusi perguruan tinggi belum optimal dalam memanfaatkannya. Selain berkembangnya sistem internal organisasi yang serba terintegrasi, tuntutan layanan berbagi berkas dan sumber daya juga sangat penting.

2. Struktur Cloud Computing Mobile.

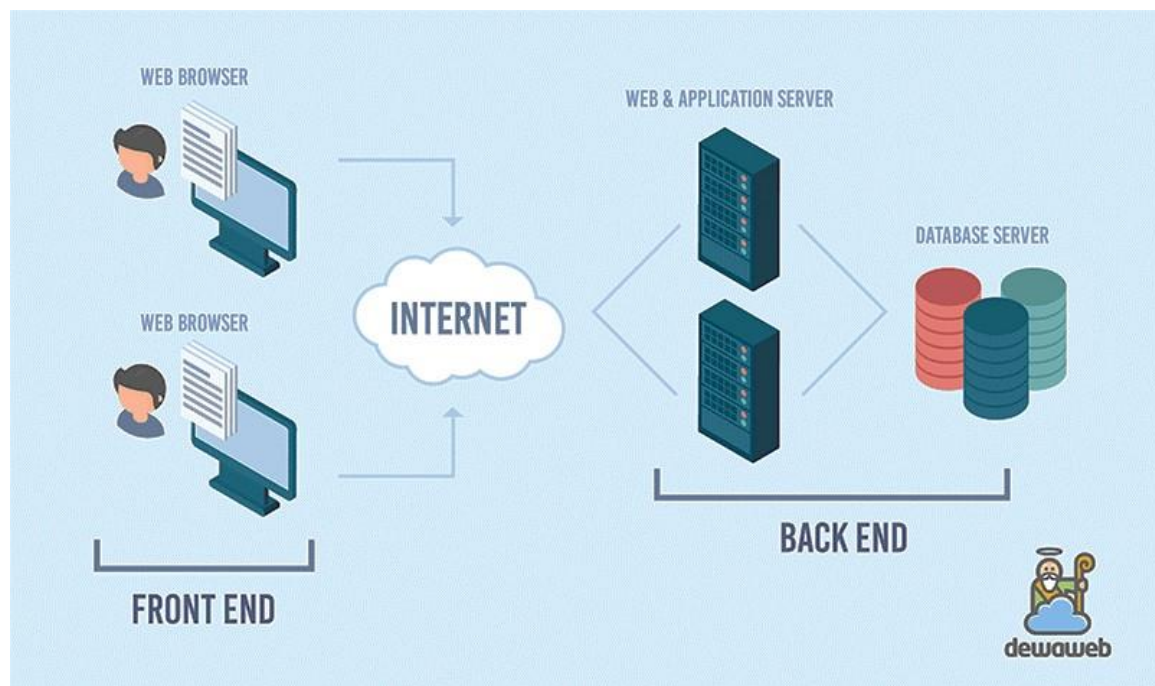
Cloud computing terdiri dari beberapa lapisan layer, dimana setiap layer berfungsi untuk mendukung komunikasi data antara setiap layanan. Struktur cloud computing terdiri dari[3] :

a. Computer Front End

Biasanya merupakan komputer desktop biasa dan muncul di halaman depan interface. Dalam hal ini Front End Cloud Computing merupakan sisi dari Client dan Cloud Computing Systemnya yang kemudian dibagi menjadi dua interface dan aplikasi yang disesuaikan dengan fungsi dan keperluan dari Cloud Computingnya.

b. Computer Back End

Computer back end adalah komputer skala besar yang biasanya berupa server computer yang dilengkapi dengan data center[5]. Biasanya, computer back end harus mempunyai kinerja yang tinggi, karena harus melayani mungkin hingga ribuan permintaan data. Perangkat yang berperan sebagai penghubung keduanya bisa berupa jaringan LAN atau internet. Sehingga dalam hal ini perangkat dapat terkoneksi satu sama lainnya untuk dapat bertukar informasi dan data[6].

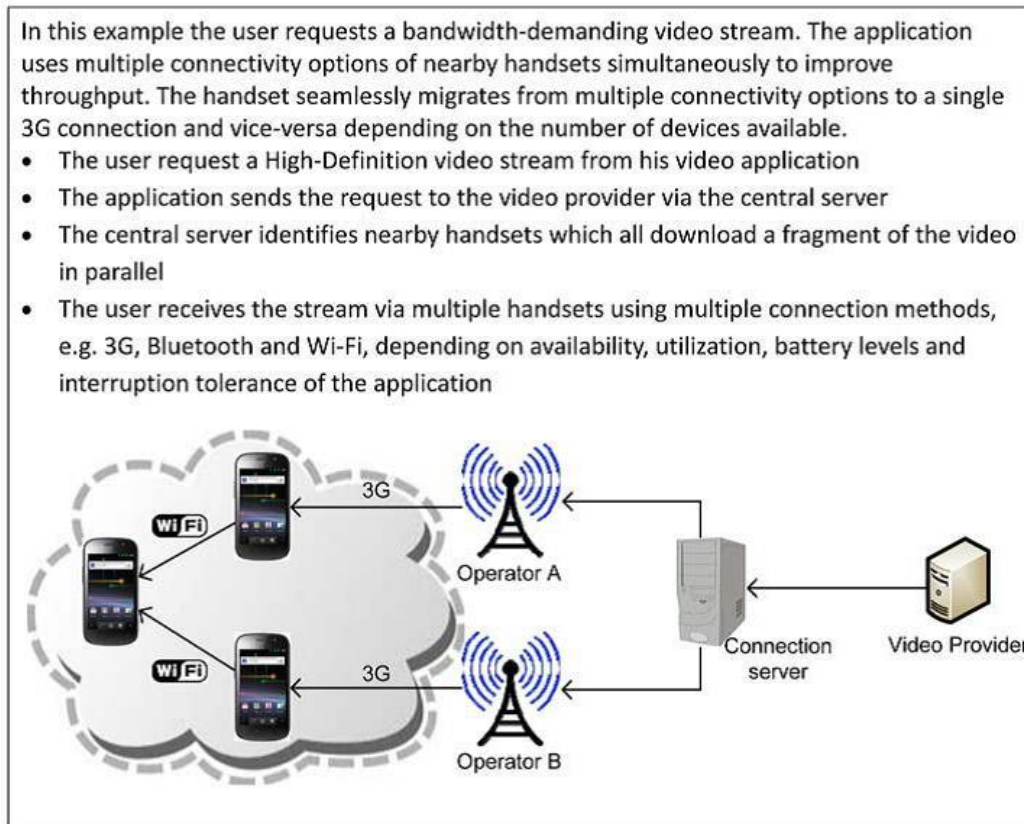


Gambar 2. Struktur Layanan Cloud Computing

3. Pemodelan Layanan Cloud Mobile

Melalui pemodelan yang dirancang memungkinkan sumber daya seluler dibagikan dan dikumpulkan dapat berubah dengan cepat karena relokasi perangkat seluler, dengan

asumsi bahwa koneksi terjadi antar perangkat seluler[7]. Setiap Pengguna dapat menggunakan cloud seluler pribadi di rumah, menggunakan cloud seluler publik saat bepergian, bekerja dan beralih ke cloud seluler lain kantor saat bekerja.



Gambar 3. Pemodelan Cloud Mobile

4. Design Docker

Aplikasi Docker adalah virtualisasi menggunakan teknologi linux container. Docker berjalan pada lapisan kernel, sehingga memungkinkan untuk menjalankan lebih banyak virtualisasi dengan sumber daya yang terbatas[9]. Aplikasi docker dipasang pada komputer server untuk mengelola kontainer yang ada. Dalam melakukan perintah dan pertukaran data dengan Aplikasi Concloud, docker mempunyai API sendiri sehingga aplikasi Concloud mengirimkan perintah dan akan mendapatkan jawabannya dalam bentuk json. Daftar rute yang digunakan dalam REST API docker dijelaskan pada tabel 1.

Tabel 1. Rute pada API Docker

No	Rute	Metode	Aksi
1	/containers/json	GET	Menampilkan semua data container yang ada
2	/containers/create	POST	Membuat container dengan image yang dimaksud
3	/containers/{id}/json	GET	Menampilkan data secara detail pada salah satu container yang dimaksud

4	/containers/{id}/start	POST	Menjalankan servis container yang dimaksud
5	/containers/{id}/stop	POST	Menghentikan servis container yang dimaksud
6	/containers/{id}/attach	POST	Menjalankan proses <i>input/output</i> pada container yang dimaksud
7	/containers/{id}	DELETE	Menghapus kontainer yang dimaksud
8	/containers/{id}/update	POST	Memperbaharui kontainer yang dimaksud
9	/images/json	GET	Menampilkan semua data <i>image</i> yang ada
10	/build	POST	Membuat image dengan file <i>tar archive</i> dengan <i>Dockerfile</i> didalamnya

5. HASIL PENELITIAN DAN PEMBAHASAN

Implementasi Docker pada Cloud Computing

Lingkungan implementasi dan pengembangan dilakukan menggunakan komputer dengan spesifikasi Host komputer adalah Intel® Core™ i5-2450M CPU @ 2.50GHz (4 CPUs) ~2.4GHz dengan memori 8 GB di Laboratorium Arsitektur dan Jaringan Komputer, Teknik Informatika ITS. Pada cloud computing, Docker akan dipasang pada satu server dengan alamat IP 10.151.36.109. Untuk API docker menggunakan port 4243[8].

a. Instalasi Docker

Instalasi docker diawali dengan mengatur repository docker terlebih dahulu. Berikut langkah-langkah:

Mempebaharui indeks paket apt

```
$ sudo apt-get update
```

Install paket-paket yang dibutuhkan apt

```
$ sudo apt-get install \ apt-transport-https \ ca-certificates \ curl \
software-properties-common
```

b. Docker's official GPG key

```
$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg |
sudo apt-key add -
```

c. Repository Docker

```
$ sudo add-apt-repository \
"deb [arch=amd64]
https://download.docker.com/linux/ubuntu
"
```

Setelah menambahkan repository docker dilanjutkan dengan proses instalasi dengan perintah , Mempebaharui indeks paket apt

```
$ sudo apt-get update
```

Install Docker

```
$ sudo apt-get install docker-ce
```

d. Network Bridge Docker Pada Cloud Seluler

Untuk mengatur jaringan container, secara pengaturan awal docker menyiapkan bridge network dengan range ip default[10]. Jika ingin membuat bridge network sendiri dalam jaringan docker maka ikuti perintah seperti dibawah ini

```
$ docker network create --driver bridge --subnet \ 10.151.36.80/28
nama_newbridge
```

Penjelasan perintah tersebut --driver untuk menentukan jenis driver bridge network, --subnet untuk menentukan subnet yang diinginkan, dan nama_newbridge adalah nama bridge yang baru dibuat dan bias diganti dengan yang lain. Agar container yang dibuat jalan pada bridge network maka saat pembuatan container perlu dimasukkan pengaturan tambahan untuk terkoneksi[11].

e. Deployment System Cloud Computing

System berbasis cloud computing yang bangun akan diimplementasikan menggunakan PHP dan framework Laravel. Fitur yang terdapat didalamnya sebagai berikut:

f. Mengelola User

Mengelola User adalah menu yang digunakan untuk mengelola data user. Fitur yang didukung antara lain adalah membuat user baru, mengubah data **user**, dan menghapus data user. Antarmuka Mengelola User ditunjukkan pada gambar 4.

Tambah User

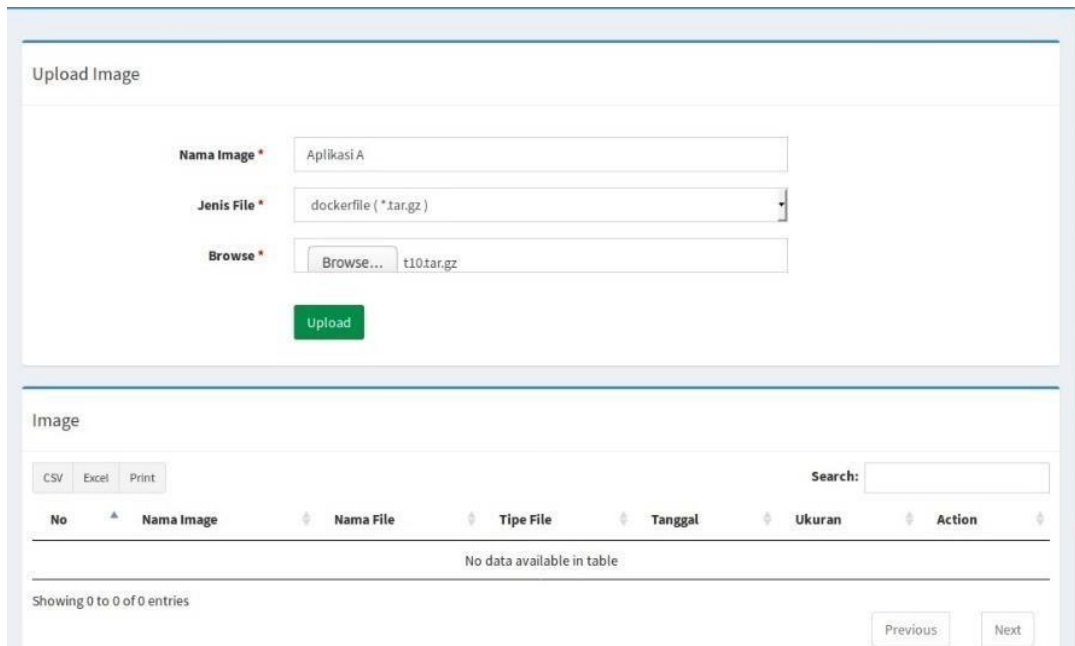
Nama *	<input type="text"/>
Tanggal Lahir *	<input type="text" value="YYYY-MM-DD"/>
Tempat Lahir *	<input type="text"/>
Alamat *	<input type="text"/>
Nomor Handphone *	<input type="text"/>
Jabatan *	<input type="text" value="admin"/>
Username *	<input type="text"/>
Password *	<input type="text"/>
Confirm Password *	<input type="text"/>

Gambar 4. Halaman Pengelolaan User

Pada gambar 4 ditunjukkan antarmuka menambah data user. Setiap data yang diisi akan disimpan ke dalam database.

g. Membuat Image

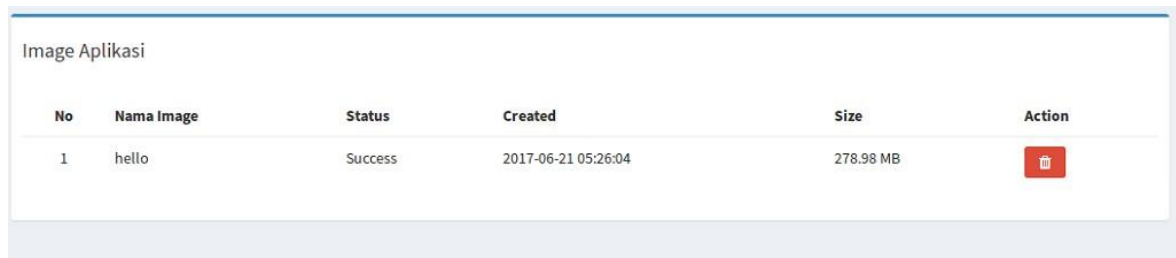
Membuat Image adalah menu yang digunakan untuk membuat image. Fitur yang didukung antara lain adalah membuat Image dan melihat log pembuatan image. Antarmuka Membuat Image ditunjukkan pada gambar 5.



Gambar 5. Halaman Membuat Image

h. Menghapus Image

Menghapus Image adalah menu yang digunakan untuk menghapus image. Fitur yang didukung antara lain adalah melihat list Image dan menghapus image. Antarmuka Menghapus Image ditunjukkan pada gambar 6.



Gambar 6. Halaman menghapus Image

i. Membuat Kontainer

Membuat kontainer adalah menu yang digunakan untuk membuat kontainer. Fitur yang didukung antara lain adalah membuat kontainer lalu menjalankannya. Antarmuka Membuat kontainer ditunjukkan pada gambar 7.

Tambah Container

Nama Image *

Nama Aplikasi *

Port *

Memory *

Environment

Gambar 7. Halaman Membuat Kontainer

j. Deploy Aplikasi

Mendeploy Aplikasi adalah menu yang digunakan untuk membuat kontainer berjalan atau berhenti. Fitur yang didukung antara lain adalah membuat container berhenti atau berjalan. Antarmuka Mendeploy aplikasi kontainer ditunjukkan pada gambar 8.

Container

CSV Excel Print Search:

No	Nama Container	IP Address	Memory Limit	Status	Size	Action
1	aplikasi_b	172.17.0.15	65536 KB	Running	64 MB	<input type="button" value="stop"/> <input type="button" value="copy"/>
2	aplikasi_a		65536 KB	Stop	64 MB	<input type="button" value="start"/> <input type="button" value="copy"/> <input type="button" value="delete"/>

Showing 1 to 2 of 2 entries

Previous **1** Next

Gambar 8. Mendeploy Aplikasi dan Menghapus Kontainer.

6. KESIMPULAN DAN SARAN

Kesimpulan

Kesimpulan yang diperoleh dalam penelitian ini adalah sebagai berikut:

Sistem cloud computing pada perangkat mobile dapat diterapkan menggunakan docker yang menghubungkan antara perangkat mobile dengan server cloud. Melalui penerapan cloud computing pada perangkat mobile data tersedia dapat digunakan secara bersama tanpa membutuhkan banyak sumberdaya fisik.

Saran

Untuk menghasilkan aplikasi yang lebih baik, penulis menganjurkan beberapa saran yaitu :

Membangun API untuk memudahkan pengguna dalam mengakses data pada server cloud computing. Menerapkan sistem keamanan pada server cloud untuk menjaga keaslian data yang diakses oleh pengguna.

DAFTAR PUSTAKA

- [1] **Cahyo Prihantoro (2017)**, Jurnal Pseudocode, Volume IV Nomor 2, September 2017, ISSN 2355-5920
- [2] **Soebhaash Dihal, Harry Bouwman, Mark de Reuver, Martijn Warnier, Christer Carlsson, (2013)**, "Mobile cloud computing: state of the art and outlook", *info*, Vol. 15 Iss: 1 pp. 4 - 16
- [3] **Mukhsin(2018)**, *TEKNOKOM Vol. 1, No. 1, Juli 2018*
- [4] **Cahyo Prihantoro(2015)**, *Seminar Nasional Teknologi Informasi dan Multimedia 2015, ISSN : 2302-3805, STMIK AMIKOM Yogyakarta, 6-8 Februari 2015*
- [5] **Anggi Srimurdianti Sukamto (2015)**, *Jurnal Edukasi dan Penelitian Informatika (JEPIN) Vol. 1, No. 1, (Juni 2015) ISSN 2460-7041*
- [6] **Sulistyo Heripracoyo (2014)**, *Analisa studi literatur manfaat implementasi komputasi awan untuk perusahaan, comtech vol. 5 no. 1 juni 2014: 154-162*
- [7] **Muhammad Faisal(208)**, "Design and Implementation of Plantation Commodities Price Information Broadcaster via Autoreply Short Message Service on Smartphone", IEEE. The 2nd East Indonesia Conference on Computer and Information Technology (EIConCIT), 2018
- [8] **Ence Achmad (2015)**, *Jurnal Informatika Polinema ISSN:2407-070X Volume: 1, No:2, Februari 2015.*
- [9] **Angga Permadi Karpriana (2015)**, *Faktor-faktor Yang Mempengaruhi Pengguna Untuk Mengadopsi Teknologi Komputasi Awan, Jurnal Akuntansi Keuangan Dan Bisnis Vol. 8, November 2015, 47-57.*
- [10] **Syaikhu (2010)**, *Komputasi Awan (Cloud Computing) Perpustakaan Pertanian*
- [11] **Abidah(2020)**, *Implementasi Sistem Basis Data Cloud Computing pada Sektor Pendidikan, Jurnal Sains dan Teknologi, Vol.1 (2), doi.org/10.24123/ saintek. v1i2.2868, 77-84, Agustus 2020*